ABSTRACT

Title of Dissertation: EXPANDING ROBUSTNESS IN RESPONSIBLE AI
FOR NOVEL BIAS MITIGATION

Samuel Dooley
Doctor of Philosophy, 2023

Dissertation Directed by: John Dickerson
Department of Computer Science

Conventional belief in the fairness community is that one should first find the highest performing model for a given problem and then apply a bias mitigation strategy. One starts with an existing model architecture and hyperparameters, and then adjusts model weights, learning procedures, or input data to make the model fairer using a pre-, post-, or in-processing bias mitigation technique. While existing methods for de-biasing machine learning systems use a fixed neural architecture and hyperparameter setting, I instead ask a fundamental question which has received little attention: *how much does model-bias arise from the architecture and hyperparameters*, and ask how can we exploit the extensive research in the fields of neural architecture search (NAS) and hyperparameter optimization (HPO) to search for more inherently fair models.

By thinking of bias mitigation in this new way, we really are expanding our conceptualization of robustness in responsible AI. Robustness is an emerging aspect of responsible AI and focuses

on maintaining model performance in the face of uncertainties and variations for all subgroups of a data population. Often robustness deals with protecting models from intentional or unintentional manipulations in data, while handling noisy or corrupted data and preserving accuracy in real-world scenarios. In other words, robustness, as commonly defined, examines the output of a system under changes to input data. However, I will broaden the idea of what robustness in responsible AI is in a manner which defines new fairness metrics, yields insights into robustness of deployed AI systems, and proposes an entirely new bias mitigation strategy.

This thesis explores the connection between robust machine learning and responsible AI. It introduces a fairness metric that quantifies disparities in susceptibility to adversarial attacks. It also audits face detection systems for robustness to common natural noises, revealing biases in these systems. Finally, it proposes using neural architecture search to find fairer architectures, challenging the conventional approach of starting with accurate architectures and applying bias mitigation strategies.

EXPANDING ROBUSTNESS IN RESPONSIBLE AI FOR NOVEL BIAS MITIGATION

by

Samuel Dooley

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2023

Advisory Committee:
Dr. John Dickerson, Chair/Advisor
Dr. Philip Resnik
Dr. Hal Daumé III
Dr. Tom Goldstein
Dr. Furong Huang
Dr. Elissa Redmiles

Acknowledgments

To my mother and father

Without the support of my family, friends, and colleagues, this work would not have been possible.

I have had the great fortune to share in the creation of this work with amazing co-authors including Vedant Nanda, Rhea Sukthanker, George Wei, Micah Goldblum, Colin White, Frank Hutter, Tom Goldstein, and John Dickerson. These collaborations have been instrumental in my progress during the PhD program.

I also worked with many other individuals on other projects throughout my time in the PhD, but with projects which did not make it into this thesis for space constraints. These projects and collaborators are listed below — thank you to everyone for their guidance and support.

I had the pleasure of working primarily with Michael Curry and John Dickerson on human value alignment in deep learning for auction design work. Specifically with Kevin Kuo, Anthony Ostuni, Elizabeth Horishny, Michael J Curry, Ping Chiang, Tom Goldstein, and John Dickerson in Kuo et al. (2020) and Neehar Peri, Michael Curry, and John Dickerson in Peri et al. (2021).

I'd also like to thank Elissa Redmiles for her guidance and collaboration on projects of deep interest to me. We worked with John Dickerson and Dana Turjeman to understand how messaging impacted Covid-19 app adoption in Louisiana (Dooley et al., 2022a). Through Elissa, I also met

Angelica Goetzen and had the pleasure of working with her on some statistical analyses of survey data on privacy sentiment (Goetzen et al., 2022).

In other user-studies, I worked closely with Michelle Mazurek on a line of work that examined the ways in which library IT staff conceived of privacy and security for their patrons. Originally devised as a class project with Michael Rosenberg, Elliot Sloate, and Sungbok Shin (Dooley et al., 2020), we worked closely with Nora McDonald and Rachel Greenstadt to devise the survey which then was conducted by me and analyzed by Alan Luo and Noel Warford (Luo et al., 2023). From these collaborations, I learned a lot about qualitative research as well as my own working style and motivations in my work. Additionally, Michael Rosenberg and I also worked on a topological data analysis project (Rawson et al., 2022) where I got to apply my undergrad math knowledge to my PhD.

I worked closely with Tom Goldstein, Micah Goldblum, and Valeriia Cherepanova on work pertaining to face recognition projects (Cherepanova et al., 2022; Dooley et al., 2021a). From them, I learned much about technical writing and research process. Also, early in my PhD studies, John introduced me to some labor market problems (Dooley and Dickerson, 2020) which I worked on with Marina Knittel (Knittel et al., 2022).

Finally, a special thanks to Colin White who has been instrumental in my progression towards the end of my program. We first started collaborating on Chapter 4 of this thesis, and this further led to an internship at Abacus.AI and ultimately a job. We've further collaborated on ForecastPFN (Khurana et al., 2023) and a bunch of forthcoming work in NAS. Thank you, Colin, for your kindness, generosity, and support.

My family has been a constant voice of support through my education, particularly my husband, mother, father, brother, sister-in-law, and niece. I particularly appreciate the constant love and encouragement of my husband as well as his family as they have cheered me on and celebrated my wins. My friends have always been there to encourage and distract me at critical times – thank you particularly to Claire and Annette. Finally, I'd like to thank all the musicians whose music I enjoyed during the many long hours and late nights working on this theses — particularly Beyoncé and her inspiring RENAISSANCE album.

In this thesis, I have been supported in part by NSF CAREER Award IIS-1846237, NIST MSE Award #20126334, NSF D-ISN Award #2039862, NSF Award CCF-1852352, NIH R01 Award NLM-013039-01, DARPA GARD #HR00112020007, DARPA SI3-CMD #S4761, DoD WHS Award #HQ003420F0035, and ARPA-E Award #4334192.

Table of Contents

List of Tables

List of Figures

He talks, he talks, how he talks, and waves
his arms.
He fills up ornate vases.
Twenty-seven an hour. And keeps the
words in with cork stoppers
(If you hold the vases to your ears you can
hear the muted syllables colliding into
each other).
I want vases, some of them ornate,
But simple ones too.
And most of them
Will have flowers

On Verbosity

Annette Ryan

Chapter 1: Introduction

Artificial Intelligence (AI) has emerged as a transformative technology with the potential to revolutionize various aspects of our lives. From personalized recommendations to autonomous vehicles, AI systems are becoming increasingly prevalent in our daily interactions. However, as AI becomes more advanced and integrated into society, concerns about its responsible use have gained significant attention.

Responsible AI refers to an ethically-informed and transparent development, deployment, and utilization of AI technologies, ensuring that they are designed and used in a manner that respects human values, rights, and well-being. The need for responsible AI arises from the potential risks associated with its widespread adoption. AI systems can inadvertently perpetuate bias, discrimination, and reinforce societal inequalities if not developed and implemented with care. For example, biased training data can lead to discriminatory outcomes, such as AI-powered hiring algorithms favoring certain demographics. Machine learning is applied to a wide variety of socially-consequential domains, e.g., credit scoring, fraud detection, hiring decisions, criminal recidivism, loan repayment, and face recognition (Mukerjee et al., 2002; Ngai et al., 2011; Learned-Miller et al., 2020; Barocas et al., 2017), with many of these applications impacting the lives of people more than ever — often in biased ways (Buolamwini and Gebru, 2018; Joo and Kärkkäinen, 2020; Wang et al., 2020b). Dozens of formal definitions of fairness have been proposed (Narayanan, 2018), and many algorithmic techniques have been developed for debiasing according to these definitions (Verma and Rubin, 2018).

Automated decision-making systems that are driven by data are being used in a variety of different real-world applications, creating the risk that these systems will perpetuate and/or create harms to people. In many cases, these systems make decisions on data points that represent humans (*e.g.*, targeted ads (Speicher et al., 2018; Ribeiro et al., 2019), personalized recommendations (Singh and Joachims, 2018; Biega et al., 2018), hiring (Schumann et al., 2019, 2020), credit scoring (Khandani et al., 2010), or recidivism prediction (Chouldechova, 2017)). In such scenarios, there is often concern regarding the fairness of outcomes of the systems (Barocas and Selbst, 2016; Galhotra et al., 2017). This has resulted in a growing body of work from Responsible

AI community that—drawing on prior legal and philosophical doctrine—aims to define, measure, and (attempt to) mitigate manifestations of unfairness in automated systems (Chouldechova, 2017; Feldman et al., 2015a; Leben, 2020; Binns, 2017).

Responsible AI aims to address such concerns by emphasizing fairness, accountability, transparency, and inclusivity in AI development and deployment processes. One crucial aspect of responsible AI is fairness. AI systems should be designed to treat all individuals fairly and without discrimination. This means avoiding bias in data collection, ensuring diverse representation during the development process, and regularly auditing AI algorithms for unintended biases. Additionally, responsible AI involves being accountable for the outcomes of AI systems. Developers and organizations should take responsibility for any harm caused by their AI technologies and implement mechanisms for redress and accountability.

Transparency is another fundamental principle of responsible AI. Users and stakeholders should have access to understandable and explainable AI systems. This means that AI algorithms should be designed in a way that allows for clear explanations of their decision-making processes. Transparent AI fosters trust, enables users to understand how AI systems work, and helps identify and rectify any potential biases or errors.

Responsible AI also emphasizes inclusivity, ensuring that the benefits and opportunities created by AI are accessible to all. This involves considering the needs and perspectives of diverse populations during AI development, addressing issues of digital divide and accessibility, and actively working towards reducing biases and disparities present in AI systems.

Another emerging aspect of Responsible AI is the robustness of systems. In traditional machine learning, robustness refers to the ability of a model to maintain its performance and generalization capabilities even in the face of uncertainties, adversarial attacks, or variations in the input data. A robust model is not only accurate on the training data but also exhibits resilience to perturbations, noise, and outliers that it may encounter during deployment. The importance of robustness arises from the fact that real-world data is often noisy, incomplete, and subject to unpredictable variations. While traditional machine learning algorithms focus on optimizing for average-case scenarios, robust machine learning aims to handle the worst-case scenarios and mitigate the risks associated with unpredictable inputs.

Robustness in machine learning encompasses various dimensions, each presenting unique challenges and trade-offs. One prominent aspect is adversarial robustness, which examines the model's vulnerability to adversarial attacks, where malicious actors deliberately manipulate the input data to deceive or mislead the model's predictions. Adversarial attacks have demonstrated the susceptibility of machine learning models to subtle perturbations that are often imperceptible to human observers. Developing models that are resistant to such attacks is crucial for security-sensitive applications. I will explore this topic in depth in Chapter 2. Most of the initial work on fairness in machine learning considered notions that were one-shot and considered the model and data distribution to be static (Zafar et al., 2019, 2017c; Chouldechova, 2017; Barocas and Selbst, 2016; Dwork et al., 2012; Zemel et al., 2013). Recently, there has been more work exploring notions of fairness that are dynamic and consider the possibility that the world (*i.e.*, the model as well as data points) might change over time (Heidari et al., 2019; Heidari and Krause, 2018; Hashimoto et al., 2018; Liu et al., 2018b). Our proposed notion of robustness bias has subtle difference from existing one-shot and dynamic notions of fairness in that it requires each partition of the population be equally robust to imperceptible changes in the input (*e.g.*, noise, adversarial perturbations, etc).

Another dimension of robustness focuses on handling noisy or corrupted data. Real-world datasets may contain outliers, missing values, or measurement errors, which can significantly impact the performance of machine learning models. Robust techniques that can effectively handle such anomalies and preserve the model's accuracy and reliability are essential.

We explore robustness to noisy or corrupted data in Chapter 3, by auditing face detection systems and show deeper and more pernicious forms of robustenss bias in these systems. Face detection identifies the presence and location of faces in images and video. Automated face detection is a core component of myriad systems—including *face recognition technologies* (FRT), wherein a detected face is matched against a database of faces, typically for identification or verification purposes. FRT-based systems are widely deployed (Hartzog, 2020; Derringer, 2019; Weise and Singer, 2020). Automated face recognition enables capabilities ranging from the relatively morally neutral (e.g., searching for photos on a personal phone (Google, 2021a)) to morally laden (e.g., widespread citizen surveillance (Hartzog, 2020), or target identification in warzones (Marson and Forrest, 2021)). Legal and social norms regarding the usage of FRT are evolving (e.g., Grother et al., 2019). For example, in June 2021, the first county-wide ban on its use for policing (see, e.g., Garvie, 2016) went into effect in the US (Gutman, 2021). Some use cases for FRT will be deemed socially repugnant and thus be either legally or *de facto* banned from use; yet, it is likely that pervasive use of facial analysis will remain—albeit with more guardrails than today (Singer, 2018).

One such guardrail that has spurred positive, though insufficient, improvements and widespread attention is the use of benchmarks. For example, in late 2019, the US National Institute of Standards and Technology (NIST) adapted its venerable Face Recognition Vendor Test (FRVT) to explicitly include concerns for demographic effects (Grother et al., 2019), ensuring such concerns propagate into industry systems. Yet, differential treatment by FRT of groups has been known for at least a decade (e.g., Klare et al., 2012; El Khiyari and Wechsler, 2016), and more recent work spearheaded by Buolamwini and Gebru (2018) uncovers unequal performance at the phenotypic subgroup level. That latter work brought widespread public, and thus burgeoning regulatory, attention to bias in FRT (e.g., Lohr, 2018; Kantayya, 2020).

One yet unexplored benchmark examines the bias present in a model's robustness (e.g., to noise, or to different lighting conditions), both in aggregate and with respect to different dimensions of the population on which it will be used. Many detection and recognition systems are not built in house, instead adapting an existing academic model or by making use of commercial cloud-based "ML as a Service" (MLaaS) platforms offered by tech giants such as Amazon, Microsoft, Google, Megvii, etc. I will present the first of its kind detailed benchmark *robustness benchmark* of six different face detection models, for fifteen types of realistic noise (Hendrycks and Dietterich, 2019), and on four well-known datasets. Across all the datasets and systems, I generally find that photos of individuals who are *older*, *masculine presenting*, of *darker skin type*, or have *dim lighting* are more susceptible to errors than their counterparts in other identities.

Addressing robustness in machine learning involves a combination of algorithmic design, feature engineering, and data preprocessing techniques. These approaches seek to make models more resilient to uncertainties and perturbations, either by introducing regularization mechanisms, utilizing ensemble methods, or leveraging domain knowledge to guide the learning process.

In this thesis, I'll broaden and deepen the connection between robust machine learning and responsible AI. In Chapter 2, I define a new fairness metric in Responsible AI which quantifies the disparity between groups with respect to how susceptible they are to adversarial attack. In Chapter 3, I audit existing academic and commercial face detection systems for their robustness to types of common natural noises. Finally, in Chapter chapter 4, I expand the conceptualization of robustness to include notions of model architecture and hyperparameters, and propose a novel bias mitigation techniques which employs neural architecture search to find more fair architectures.

Conventional wisdom is that in order to effectively mitigate bias, we should start by selecting a model architecture and set of hyperparameters which are optimal in terms of accuracy and then apply a mitigation strategy to reduce bias while minimally impacting accuracy. As datasets become larger and training becomes more computationally intensive, especially in the case of computer vision and natural language processing, it is becoming increasingly more common in applications to start with a very large pretrained model, and then fine-tune for the specific use-case (Chi et al., 2017; Käding et al., 2016; Ouyang et al., 2016; Too et al., 2019). While existing methods for de-biasing machine learning systems use a fixed neural architecture and hyperparameter setting, I instead ask a fundamental question which has received little attention: *how much does model-bias arise from the architecture and hyperparameters?* I further ask whether we can we exploit the extensive research in the fields of neural architecture search (NAS) (Elsken et al., 2019) and hyperparameter optimization (HPO) (Feurer and Hutter, 2019) to search for more inherently fair models.

Many debiasing algorithms fit into one of three (or arguably four (Savani et al., 2020)) categories: pre-processing (e.g., Feldman et al., 2015b; Ryu et al., 2018; Quadrianto et al., 2019; Wang and Deng, 2020), in-processing (e.g., Zafar et al., 2017b, 2019; Donini et al., 2018; Goel et al., 2018; Padala and Gujar, 2020; Wang and Deng, 2020; Martinez et al., 2020; Nanda et al., 2021; Diana et al., 2020; Lahoti et al., 2020), or post-processing (e.g., Hardt et al., 2016; Wang et al., 2020b). I, however, pose the simple question, what if these approaches are using an architecture which is inherently less fair than another architecture? To explore this topic, I employ neural architecture search.

Neural architecture search (NAS) is a field of research that focuses on automating the design and optimization of neural network architectures. It aims to discover or construct neural network structures that achieve high performance on various tasks while reducing the manual effort required for architecture engineering. Traditionally, neural network architectures were designed by human experts based on their domain knowledge and intuition. However, as deep learning models have grown in complexity and scale, manually designing architectures that yield optimal performance has become increasingly challenging and time-consuming. NAS approaches employ various strategies to automatically explore the vast search space of possible architectures. One common technique is to use reinforcement learning or evolutionary algorithms to iteratively generate, evaluate, and refine architectures. These algorithms leverage performance feedback from the training process to guide the search towards architectures with improved performance. NAS algorithms often incorporate additional components like performance predictors or surrogate models, which help estimate the performance of unseen architectures based on their characteristics. These models aid in efficiently exploring the architecture space and reduce the computational cost associated with evaluating each architecture.

Motivated by the belief that the inductive bias of a model architecture is more important than the bias mitigation strategy, I'll use NAS to take a different approach to bias mitigation. We show that finding an *a*rchitecture that is more fair offers significant gains compared to conventional bias mitigation strategies in the domain of face recognition, a task that is notoriously challenging to de-bias. To this end, I'll conduct the first neural architecture search for fairness, jointly with a search for hyperparameters. Our search outputs a suite of models which Pareto-dominate all other competitive architectures in terms of accuracy and fairness on the two most widely used datasets for face identification: CelebA and VGGFace2. This work challenges the assumption that bias mitigation pipelines should default to existing popular architectures which were optimized for accuracy — instead I'll show that it may be more beneficial to begin with a fairer architecture as the foundation of such pipelines.

You gave up on expecting things to make
sense
Knew at some point this would not be a
puzzle you would ever complete

So you learn to hold it all
bodies that are home
and how the days unfold
one after the next
A continuous scroll of doing your best
and trusting
what the darkness will hold

Call it Love

Jena Schwartz

Chapter 2: Adversarial Robustness

This work was done in collaboration with my co-first author Vedant Nanda, as well as Sahil Singla, John P. Dickerson, and Soheil Feizi, and was presented at FAccT, 2021 (Nanda et al., 2021).

Deep neural networks (DNNs) are increasingly used in real-world applications (e.g. facial recognition). This has resulted in concerns about the fairness of decisions made by these models.

Various notions and measures of fairness have been proposed to ensure that a decision-making system does not disproportionately harm (or benefit) particular subgroups of the population. In this chapter, we argue that traditional notions of fairness that are only based on models' outputs are not sufficient when the model is vulnerable to adversarial attacks. We argue that in some cases, it may be easier for an attacker to target a particular subgroup, resulting in a form of *robustness bias*. We show that measuring robustness bias is a challenging task for DNNs and propose two methods to measure this form of bias. We then conduct an empirical study on state-of-the-art neural networks on commonly used real-world datasets such as CIFAR-10, CIFAR-100, Adience, and UTKFace and show that in almost all cases there are subgroups (in some cases based on sensitive attributes like race, gender, etc) which are less robust and are thus at a disadvantage. We argue that this kind of bias arises due to both the data distribution and the highly complex nature of the learned decision boundary in the case of DNNs, thus making mitigation of such biases a non-trivial task. Our results show that robustness bias is an important criterion to consider while auditing real-world systems that rely on DNNs for decision making.

2.1 Introduction

Automated decision-making systems that are driven by data are being used in a variety of different real-world applications. In many cases, these systems make decisions on data points that represent humans (*e.g.*, targeted ads (Speicher et al., 2018; Ribeiro et al., 2019), personalized recommendations (Singh and Joachims, 2018; Biega et al., 2018), hiring (Schumann et al., 2019, 2020), credit scoring (Khandani et al., 2010), or recidivism prediction (Chouldechova, 2017)). In such scenarios, there is often concern regarding the fairness of outcomes of the systems (Barocas and Selbst, 2016; Galhotra et al., 2017). This has resulted in a growing body of work from the nascent Fairness, Accountability, Transparency, and Ethics (FATE) community that—drawing on prior legal and philosophical doctrine—aims to define, measure, and (attempt to) mitigate manifestations of unfairness in automated systems (Chouldechova, 2017; Feldman et al., 2015a; Leben, 2020; Binns, 2017).

Most of the initial work on fairness in machine learning considered notions that were one-shot and considered the model and data distribution to be static (Zafar et al., 2019, 2017c; Chouldechova, 2017; Barocas and Selbst, 2016; Dwork et al., 2012; Zemel et al., 2013). Recently, there has been more work exploring notions of fairness that are dynamic and consider the possibility that the world (*i.e.*, the model as well as data points) might change over time (Heidari et al., 2019; Heidari and Krause, 2018; Hashimoto et al., 2018; Liu et al., 2018b). Our proposed notion of robustness bias has subtle difference from existing one-shot and dynamic notions of fairness in that it requires each partition of the population be equally robust to imperceptible changes in the input (*e.g.*, noise, adversarial perturbations, etc).

We propose a simple and intuitive notion of *robustness bias* which requires subgroups of populations to be equally "robust." Robustness can be defined in multiple different ways (Szegedy et al., 2014; Goodfellow et al., 2015; Papernot et al., 2016). We take a general definition which assigns points that are farther away from the decision boundary higher robustness. Our key contributions are as follows:

- We define a simple, intuitive notion of ***robustness bias*** that requires all partitions of the dataset to be equally robust. We argue that such a notion is especially important when the decision-making system is a deep neural network (DNN) since these have been shown to be susceptible to various attacks (Carlini and Wagner, 2017; Moosavi-Dezfooli et al., 2016). Importantly, our notion depends not only on the outcomes of the system, but also on the distribution of distances of data-points from the decision boundary, which in turn is a characteristic of *both* the data distribution and the learning process.

- We propose different methods to ***measure this form of bias***. Measuring the exact distance of a point from the decision boundary is a challenging task for deep neural networks which have a highly non-convex decision boundary. This makes the measurement of robustness bias a non-trivial task. In this chapter we leverage the literature on adversarial machine learning and show that we can efficiently approximate robustness bias by using adversarial attacks and randomized smoothing to get estimates of a point's distance from the decision boundary.

- We do an in-depth analysis of *robustness bias* on popularly used datasets and models. Through ***extensive empirical evaluation*** we show that unfairness can exist due to different partitions of a dataset being at different levels of robustness for many state-of-the art models that are trained on common classification datasets. We argue that this form of unfairness can happen due to both the data distribution and the learning process and is an important criterion to consider when auditing models for fairness.

2.1.1 Related Work

Fairness in ML. Models that learn from historic data have been shown to exhibit unfairness, *i.e.*, they disproportionately benefit or harm certain subgroups (often a sub-population that shares a common sensitive attribute such as race, gender *etc.*) of the population (Barocas and Selbst, 2016;

Chouldechova, 2017; Khandani et al., 2010). This has resulted in a lot of work on quantifying, measuring and to some extent also mitigating unfairness (Dwork et al., 2012; Dwork and Ilvento, 2018; Zemel et al., 2013; Zafar et al., 2019, 2017c; Hardt et al., 2016; Grgić-Hlača et al., 2018; Adel et al., 2019; Wadsworth et al., 2018; Saha et al., 2020; Donini et al., 2018; Calmon et al., 2017; Kusner et al., 2017; Kilbertus et al., 2017; Pleiss et al., 2017; Wang et al., 2020b). Most of these works consider notions of fairness that are one-shot—that is, they do not consider how these systems would behave over time as the world (*i.e.*, the model and data distribution) evolves. Recently more works have taken into account the dynamic nature of these decision-making systems and consider fairness definitions and learning algorithms that fare well across multiple time steps (Heidari et al., 2019; Heidari and Krause, 2018; Hashimoto et al., 2018; Liu et al., 2018b). We take inspiration from both the one-shot and dynamic notions, but take a slightly different approach by requiring all subgroups of the population to be equally robust to minute changes in their features. These changes could either be random (*e.g.*natural noise in measurements) or carefully crafted adversarial noise. This is closely related to Heidari et al. (2019)'s effort-based notion of fairness; however, their notion has a very specific use case of societal scale models whereas our approach is more general and applicable to all kinds of models. Our work is also closely related to and inspired by Zafar et al.'s use of a regularized loss function which captures fairness notions and reduces disparity in outcomes (Zafar et al., 2019). There are major differences in both the *approach* and *application* between our work and that of Zafar et al's. Their disparate impact formulation aims to equalize the average distance of points to the decision boundary, $\mathbb{E}[d(x)]$; our approach, instead, aims to equalize the number of points that are "safe", i.e., $\mathbb{E}[\mathbb{1}\{d(x) > \tau\}]$ (see section 2.3 for a detailed description). Our proposed metric is preferable for applications of adversarial attack or noisy data, the focus of our paper; whereas the metric of

Zafar et al is more applicable for an analysis of the consequence of a decision in a classification setting.

Robustness. Deep Neural Networks (DNNs) have been shown to be susceptible to carefully crafted adversarial perturbations which—imperceptible to a human—result in a misclassification by the model (Szegedy et al., 2014; Goodfellow et al., 2015; Papernot et al., 2016). In the context of our paper, we use adversarial attacks to approximate the distance of a data point to the decision boundary. For this we use state-of-the-art white-box attacks proposed by Moosavi-Dezfooli et al. (2016) and Carlini and Wagner (2017). Due to the many works on adversarial attacks, there have been many recent works on provable robustness to such attacks. The high-level goal of these works is to estimate a (tight) lower bound on the distance of a point from the decision boundary (Cohen et al., 2019; Salman et al., 2019; Singla and Feizi, 2020). We leverage these methods to estimate distances from the decision boundary which helps assess robustness bias (defined formally in Section 2.3).

Fairness and Robustness. Recent works have proposed poisoning attacks on fairness (Solans et al., 2020; Mehrabi et al., 2020). Khani and Liang (2019) analyze why noise in features can cause disparity in error rates when learning a regression. We believe that our work is the very first to show that different subgroups of the population can have different levels of robustness which can lead to unfairness. We hope that this will lead to more work at the intersection of these two important sub fields of ML.

2.2 Heterogeneous Robustness

In a classification setting, a learner is given data $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ consisting of inputs $x_i \in \mathbb{R}^d$ and outputs $y_i \in \mathcal{C}$ which are labels in some set of classes $\mathcal{C} = \{c_1, \ldots, c_k\}$. These classes form a partition on the dataset such that $\mathcal{D} = \bigsqcup_{c \in \mathcal{C}} \{(x_i, y_i) \mid y_i = c_j\}$. The goal of learning in decision boundary-based optimization is to draw delineations between points in feature space which sort the data into groups according to their class label. The learning generally tries to maximize the classification accuracy of the decision boundary choice. A learner chooses some loss function $\mathcal{L}$ to minimize on a training dataset, parameterized by parameters θ, while maximizing the classification accuracy on a test dataset.

Of course there are other aspects to classification problems that have recently become more salient in the machine learning community. Considerations about the fairness of classification decisions, for example, are one such way in which additional constraints are brought into a

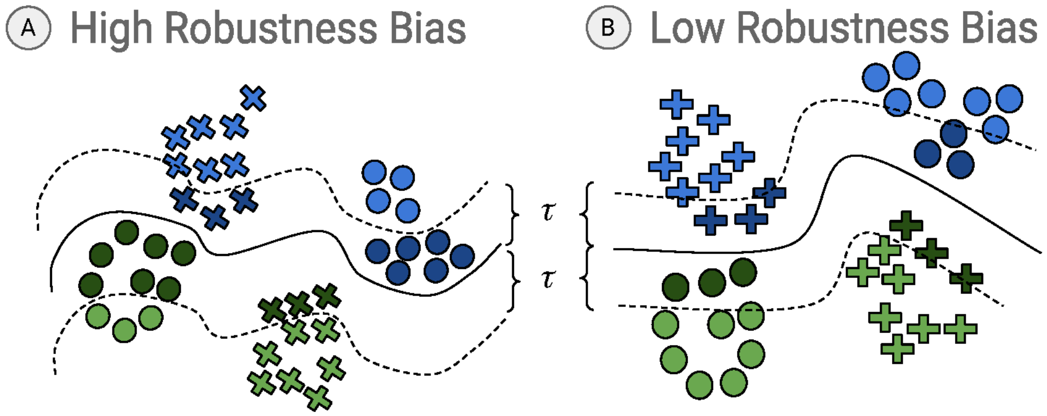
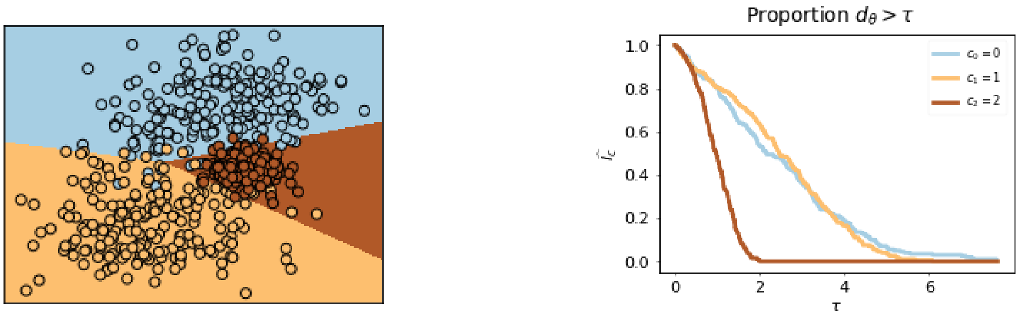


Figure 2.1: A toy example showing robustness bias. A.) the classifier (solid line) has 100% accuracy for blue and green points. However for a budget τ (dotted lines), 70% of points belonging to the "round" subclass (showed by dark blue and dark green) will get attacked while only 30% of points in the "cross" subclass will be attacked. This shows a clear bias against the "round" subclass which is less robust in this case. B.) shows a different classifier for the same data points also with 100% accuracy. However, in this case, with the same budget τ, 30% of both "round" and "cross" subclass will be attacked, thus being less biased.



(a) Three-class classification problem for randomly generated data.

(b) Proportion samples which are greater than τ away from a decision boundary.

Figure 2.2: An example of multinomial logistic regression.

learner's optimization strategy. In these settings, the data $\mathcal{D} = \{(x_i, y_i, s_i)\}_{i=1}^N$ is imbued with some metadata which have a sensitive attribute $\mathcal{S} = \{s_1, \ldots, s_t\}$ associated with each point. Like the classes above, these sensitive attributes form a partition on the data such that $\mathcal{D} = \bigsqcup_{s \in \mathcal{S}} \{(x_i, y_i, s_i) \mid s_i = s\}$. Without loss of generality, we assume a single sensitive attribute. Generally speaking, learning with fairness in mind considers the output of a classifier based off of the partition of data by the sensitive attribute, where some objective behavior, like minimizing disparate impact or treatment (Zafar et al., 2019), is integrated into the loss function or learning procedure to find the optimal parameters θ.

There is not a one-to-one correspondence between decision boundaries and classifier performance. For any given performance level on a test dataset, there are infinitely many decision boundaries which produce the same performance, see Figure 2.1. This raises the question: *if we consider all decision boundaries or model parameters which achieve a certain performance, how do we choose among them? What are the properties of a desirable, high-performing decision boundary?* As the community has discovered, one *undesirable* characteristic of a decision boundary is its proximity to data which might be susceptible to adversarial attack (Goodfellow et al., 2015; Szegedy et al., 2014; Papernot et al., 2016). This provides intuition that we should prefer boundaries that are as far away as possible from example data (Suykens and Vandewalle, 1999; Boser et al., 1992).

Let us look at how this plays out in a simple example. In multinomial logistic regression, the decision boundaries are well understood and can be written in closed form. This makes it easy for us to compute how close each point is to a decision boundary. Consider for example a dataset and learned classifier as in Figure 2.2a. For this dataset, we observe that the brown class, as a whole, is closer to a decision boundary than the yellow or blue classes. We can quantify this by plotting the proportion of data that are greater than a distance τ away from a decision boundary, and then varying τ. Let $d_\theta(x)$ be the minimal distance between a point x and a decision boundary corresponding to parameters θ. For a given partition $\mathcal{P}$ of a dataset, $\mathcal{D}$, such that $\mathcal{D} = \bigsqcup_{P \in \mathcal{P}} P$, we define the function:

$$\widehat{I_P}(\tau) = \frac{|\{(x, y) \in P \mid d_\theta(x) > \tau, y = \hat{y}\}|}{|P|}$$

If each element of the partition is uniquely defined by an element, say a class label, c, or a sensitive attribute label, s, we equivalently will write $\widehat{I_c}(\tau)$ or $\widehat{I_s}(\tau)$ respectively. We plot this over a range of τ in Figure 2.2b for the toy classification problem in Figure 2.2a. Observe that the function for the brown class decreases significantly faster than the other two classes, quantifying how much closer the brown class is to the decision boundary.

From a strictly classification accuracy point of view, the brown class being significantly closer to the decision boundary is not of concern; all three classes achieve similar classification accuracy. However, when we move away from this toy problem and into neural networks on real data, this difference between the classes could become a potential vulnerability to exploit,

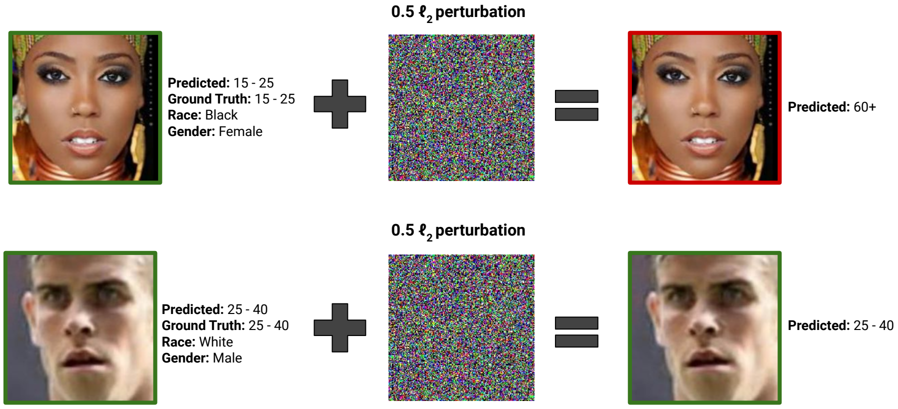


Figure 2.3: An example of robustness bias in the UTKFace dataset. A model trained to predict age group from faces is fooled for an inputs belonging to certain subgroups (black and female in this example) for a given perturbation, but is robust for inputs belonging to other subgroups (white and male in this example) for the *same magnitude* of perturbation. We use the UTKFace dataset to make a broader point that robustness bias can cause harms. In the specific case of UTKFace (and similar datasets), the task definition of predicting age from faces itself is flawed, as has been noted in many previous studies (Cramer et al., 2019; Crawford and Paglen, 2019; Buolamwini and Gebru, 2018).

particularly when we consider adversarial examples.

2.3 Robustness Bias

Our goal is to understand how susceptible different classes are to perturbations (e.g., natural noise, adversarial perturbations). Ideally, no one class would be more susceptible than any other, but this may not be possible. We have observed that for the same dataset, there may be some classifiers which have differences between the distance of that partition to a decision boundary; and some which do not. There may also be one partition $\mathcal{P}$ which exhibits this discrepancy, and another partition $\mathcal{P}'$ which does not. Therefore, we make the following statement about robustness bias:

Definition 1. A dataset $\mathcal{D}$ with a partition $\mathcal{P}$ and a classifier parameterized by θ exhibits **robustness**

bias if there exists an element $P \in \mathcal{P}$ for which the elements of P are either significantly closer to (or significantly farther from) a decision boundary than elements not in P.

A partition $\mathcal{P}$ may be based on sensitive attributes such as race, gender, or ethnicity—or other class labels. For example, given a classifier and dataset with sensitive attribute “race”, we might say that classifier exhibits robustness bias if, partitioning on that sensitive attribute, for some value of “race” the average distance of members of that particular racial value are substantially closer to the decision boundary than other members.

We might say that a dataset, partition, and classifier do not exhibit robustness bias if for all $P, P' \in \mathcal{P}$ and all $\tau > 0$

$$\begin{gathered}\mathbb{P}_{(x,y)\in\mathcal{D}}\{d_\theta(x) > \tau \mid x \in P, y = \hat{y}\} \approx \\ \mathbb{P}_{(x,y)\in\mathcal{D}}\{d_\theta(x) > \tau \mid x \in P', y = \hat{y}\}.\end{gathered} \tag{2.1}$$

Intuitively, this definition requires that for a given perturbation budget τ and a given partition P, one should not have any incentive to perturb data points from P over points that do not belong to P. Even when examining this criteria, we can see that this might be particularly hard to satisfy. Thus, we want to quantify the disparate susceptibility of each element of a partition to adversarial attack, i.e., how much farther or closer it is to a decision boundary when compared to all other points. We can do this with the following function for a dataset $\mathcal{D}$ with partition element $P \in \mathcal{P}$ and classifier parameterized by θ:

$$\begin{aligned}RB(P, \tau) = \ &|\,\mathbb{P}_{x\in\mathcal{D}}\{d_\theta(x) > \tau \mid x \in P, y = \hat{y}\} - \\ &\mathbb{P}_{x\in\mathcal{D}}\{d_\theta(x) > \tau \mid x \notin P, y = \hat{y}\}\,|\end{aligned} \tag{2.2}$$

Observe that $RB(P, \tau)$ is a large value if and only if the elements of P are much more (or

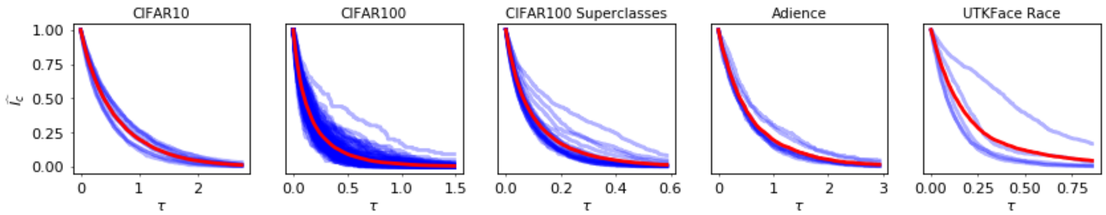


Figure 2.4: For each dataset, we plot $\widehat{I_c}(\tau)$ for each class c in each dataset. Each blue line represents one class. The red line represents the mean of the blue lines, i.e., $\sum_{c \in \mathcal{C}} \widehat{I_c}(\tau)$ for each τ.

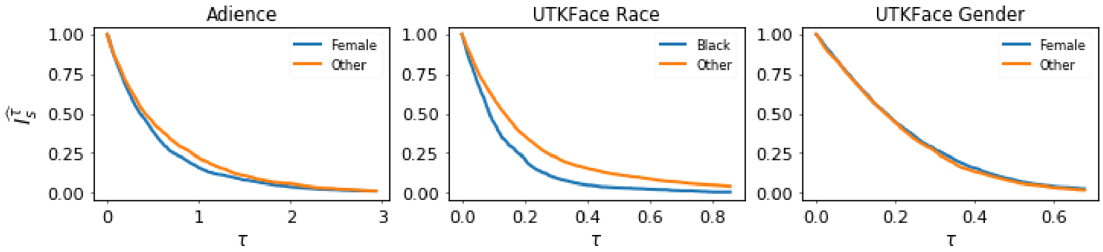


Figure 2.5: For each dataset, we plot $\widehat{I_s^\tau}$ for each sensitive attribute s in each dataset.

less) adversarially robust than elements not in P. We can then quantify this for each element $P \in \mathcal{P}$—but a more pernicious variable to handle is τ. We propose to look at the area under the curve $\widehat{I_P}$ for all τ:

$$\sigma(P) = \frac{AUC(\widehat{I_P}) - AUC(\sum_{P' \neq P} \widehat{I_{P'}})}{AUC(\sum_{P' \neq P} \widehat{I_{P'}})} \tag{2.3}$$

Note that these notions take into account the distances of data points from the decision boundary and hence are orthogonal and complementary to other traditional notions of bias or fairness (*e.g.*, disparate impact/disparate mistreatment (Zafar et al., 2019), etc). This means that having lower robustness bias does not necessarily come at the cost of fairness as measured by these notions. Consider the motivating example shown in Figure 2.1: the decision boundary on the right has lower robustness bias but preserves all other common notions (*e.g.* (Hardt et al., 2016; Dwork et al., 2012; Zafar et al., 2017c)) as both classifiers maintain 100% accuracy.

2.3.1 Real-world Implications: Degradation of Quality of Service

Deep neural networks are the core of many real world applications, for example, facial recognition, object detection, etc. In such cases, perturbations in the input can occur due to multiple factors such as noise due to the environment or malicious intent by an adversary. Previous works have highlighted how harms can be caused due to the degradation in quality of service for certain sub-populations (Cramer et al., 2019; Holstein et al., 2019). Figure 2.3 shows an example of inputs from the UTKFace dataset where an ℓ_2 perturbation of 0.5 could change the predicted label for an input with race "black" and gender "female" but an input with race "white" and gender "male" was robust to the same magnitude of perturbation. In such a case, the system worked better for a certain sub-group (white, male) thus resulting in unfairness. It is important to note that we use datasets such as Adience and UTKFace (described in detail in section 2.5) only to demonstrate the importance of having unbiased robustness. As noted in previous works, the very task of predicting age from a person's face is a flawed task definition with many ethical concerns (Cramer et al., 2019; Buolamwini and Gebru, 2018; Crawford and Paglen, 2019).

2.4 Measuring Robustness Bias

Robustness bias as defined in the previous section requires a way to measure the distance between a point and the (closest) decision boundary. For deep neural networks in use today, a direct computation of $d_\theta(x)$ is not feasible due to their highly complicated and non-convex decision boundary. However, we show that we can leverage existing techniques from the literature on adversarial attacks to efficiently approximate $d_\theta(x)$. We describe these in more detail in this section.

2.4.1 Adversarial Attacks (Upper Bound)

For a given input and model, one can compute an *upper bound* on $d_\theta(x)$ by performing an optimization which alters the input image slightly so as to place the altered image into a different category than the original. Assume for a given data point x, we are able to compute an adversarial image $\tilde{x}$, then the distance between these two images provides an upper bound on distance to a decision boundary, i.e, $\|x - \tilde{x}\| \geq d_\theta(x)$.

We evaluate two adversarial attacks: DeepFool (Moosavi-Dezfooli et al., 2016) and Carlini-Wagner's L2 attack (Carlini and Wagner, 2017). We extend $\widehat{I_P}$ for DeepFool and CarliniWagner as

$$\widehat{I_P^{DF}} = \frac{|\{(x,y) \in P | \tau < \|x - \tilde{x}\|, y = \hat{y}\}|}{|P|} \tag{2.4}$$

and

$$\widehat{I_P^{CW}} = \frac{|\{(x,y) \in P | \tau < \|x - \tilde{x}\|, y = \hat{y}\}|}{|P|} \tag{2.5}$$

respectively. We use similar notation to define $\sigma^{DF}(P)$, and $\sigma^{CW}(P)$ (σ as defined in Eq 2.3). While these methods are guaranteed to yield upper bounds on $d_\theta(x)$, they need not yield similar behavior to $\widehat{I_P}$ or $\sigma(P)$. We perform an evaluation of this in Section 2.7.1.

2.4.2 Randomized Smoothing (Lower Bound)

Alternatively one can compute a lower bound on $d_\theta(x)$ using techniques from recent works on training provably robust classifiers (Salman et al., 2019; Cohen et al., 2019). For each input, these methods calculate a radius in which the prediction of x will not change (i.e. the robustness certificate). In particular, we use the *randomized smoothing* method (Cohen et al., 2019; Salman et al., 2019) since it is scalable to large and deep neural networks and leads to the state-of-the-art in provable defenses. Randomized smoothing transforms the base classifier f to a new smooth classifier g by averaging the output of f over noisy versions of x. This new classifier g is more robust to perturbations while also having accuracy on par to the original classifier. It is also possible to calculate the radius δ_x (in the ℓ_2 distance) in which, with high probability, a given input's prediction remains the same for the smoothed classifier (i.e. $d_\theta(x) \geq \delta_x$). A given input x is then said to be provably robust, with high probability, for a δ_x ℓ_2-perturbation where δ_x is the robustness certificate of x.

For each point we use its δ_x, calculated using the method proposed by Salman et al. (2019), as a proxy for $d_\theta(x)$. The magnitude of δ_x for an input is a measure of how robust an input is. Inputs with higher δ_x are more robust than inputs with smaller δ_x. Again, we extend $\widehat{I_P}$ for Randomized Smoothing as

$$\widehat{I_P^{RS}} = \frac{|\{(x,y) \in P|\tau < \delta_x, y = \hat{y}\}|}{|P|} \tag{2.6}$$

We use similar notation to define $\sigma^{RS}(P)$ (see Eq 2.3).

2.5 Empirical Evidence of Robustness Bias in the Wild

We hypothesize that there exist datasets and model architectures which exhibit robustness bias. To investigate this claim, we examine several image-based classification datasets and common model architectures.

2.5.0.1 Datasets and Model Architectures:

We perform these tests of the datasets **CIFAR-10** (Krizhevsky, 2009), **CIFAR-100** (Krizhevsky, 2009) (using both 100 classes and 20 super classes), **Adience** (Eidinger et al., 2014), and **UTK-Face** (Zhang et al., 2017). The first two are widely accepted benchmarks in image classification, while the latter two provide significant metadata about each image, permitting various partitions of the data by final classes and sensitive attributes.

CIFAR-10, CIFAR-100, CIFAR100Super. These are standard deep learning benchmark datasets. Both CIFAR-10 and CIFAR-100 contain $60,000$ images in total which are split into $50,000$ train and $10,000$ test images. The task is to classify a given image. Images are mean normalized with mean and std of $(0.5, 0.5, 0.5)$.

UTKFace. Contains images of a people labeled with race, gender and age. We split the dataset into a random $80:20$ train:test split to get $4,742$ test and $18,966$ train samples. We bin the age into 5 age groups and convert this into a 5-class classification problem. Images are normalized with mean and std of 0 and 1 respectively.

Adience. Contains images of a people labeled with gender and age group. Task is to classify a given image into one of 8 age groups. We split the dataset into a random 80:20 train:test split to get $14,007$ train and $3,445$ test samples. Images are normalized with mean and std of $(0.485, 0.456, 0.406)$ and $(0.229, 0.224, 0.225)$ respectively.

Our experiments were performed using PyTorch's torchvision module (Paszke et al., 2019). We first explore a simple **Multinomial Logistic Regression** model which could be fully analyzed with direct computation of the distance to the nearest decision boundary. For convolutional neural networks, we focus on **Alexnet** (Krizhevsky, 2014), **VGG19** (Simonyan and Zisserman, 2015), **ResNet50** (He et al., 2016a), **DenseNet121** (Huang et al., 2017), and **Squeezenet1_0** (Iandola et al., 2016) which are all available through torchvision. We use these models since these are widely used for a variety of tasks. We achieve performance that is comparable to state of the art performance on these datasets for these models. Additionally we also train some other popularly used dataset specific architectures like a deep convolutional neural network (we call this **Deep CNN**)[1] and **PyramidNet** ($\alpha = 64$, depth=110, no bottleneck) (Han et al., 2017) for CIFAR-10. We re-implemented Deep CNN in pytorch and used the publicly available repo to train PyramidNet[2]. We use another deep convolutional neural network (which we refer to as **Deep CNN CIFAR100**[3] and **PyramidNet** ($\alpha = 48$, depth=164, with bottleneck) for CIFAR-100 and CIFAR-100Super. For Adience and UTKFace we additionally take simple deep convolutional neural networks with multiple convolutional layers each of which is followed by a ReLu activation, dropout and maxpooling. As opposed to architectures from torchvision (which are pre-trained on ImageNet) these architectures are trained from scratch on the respective datasets. We refer to them as **UTK Classifier** and **Adience Classifier** respectively. These simple models serve two purposes: they form reasonable baselines for comparison with pre-trained ImageNet models finetuned on the respective datasets, and they allow us to analyze robustness bias when models are trained from scratch.

Accuracy of models trained on the datasets can be found in Table 2.1.

In Sections 2.7 and 2.8 we audit these datasets and the listed models for robustness bias. In section 2.6, we train logistic regression on all the mentioned datasets and evaluate robustness

[1] http://torch.ch/blog/2015/07/30/cifar.html

[2] https://github.com/dyhan0920/PyramidNet-PyTorch

[3] https://github.com/aaron-xichen/pytorch-playground/blob/master/cifar/model.py

Table 2.1: Test data performance of all models on different datasets.

	Deep CNN (CIFAR100)	**PyramidNet**	**Adience Classifier**	**UTK Classifier**	**Resnet50**	**Alexnet**	**VGG**	**Densenet**	**Squeeze-net**
Adience	-	-	48.80	-	49.75	46.04	51.41	50.80	49.49
UTKFace	-	-	-	66.25	69.82	68.09	69.89	69.15	70.73
CIFAR10	86.97	86.92	-	-	83.26	92.08	89.53	85.17	76.97
CIFAR100	59.60	56.42	-	-	55.81	71.31	64.39	61.05	40.36
CIFAR100super	71.78	67.55	-	-	67.27	80.7	76.06	71.22	55.16

bias using an exact computation. We then show in section 2.7 and 2.8 that robustness bias can be efficiently approximated using the techniques mentioned in 2.4.1 and 2.4.2 respectively for much more complicated models, which are often used in the real world. We also provide a thorough analysis of the types of robustness biases exhibited by some of the popularly used models on these datasets.

2.6 Exact Computation in a Simple Model: Multinomial Logistic Regression

We begin our analysis by studying the behavior of multinomial logistic regression. Admittedly, this is a simple model compared to modern deep-learning-based approaches; however, it enables is to explicitly compute the exact distance to a decision boundary, $d_\theta(x)$. We fit a regression to each of our vision datasets to their native classes and plot $\widehat{I_c}(\tau)$ for each dataset. Figure 2.2 shows the distributions of $\widehat{I_c}(\tau)$, from which we observe three main phenomena: (1) the general shape of the curves are similar for each dataset, (2) there are classes which are significant outliers from the other classes, and (3) the range of support of the τ for each dataset varies significantly. We discuss each of these individually.

First, we note that the shape of the curves for each dataset is qualitatively similar. Since the form of the decision boundaries in multinomial logistic regression are linear delineations in the input space, it is fair to assume that this similarity in shape in Figure 2.4 can be attributed to the nature of the classifier.

Second, there are classes c which indicate disparate treatment under $\widehat{I_c}(\tau)$. The treatment disparities are most notable in UTKFace, the superclass version CIFAR-100, and regular CIFAR-100. This suggests that, when considering the dataset as a whole, these outlier classes are less suceptible to adversarial attack than other classes. Further, in UTKFace, there are some classes that are considerably more susceptible to adversarial attack because a larger proportion of that class is closer to the decision boundaries.

We also observe that the median distance to decision boundary can vary based on the dataset. The median distance to a decision boundary for each dataset is: 0.40 for CIFAR-10; 0.10 for CIFAR-100; 0.06 for the superclass version of CIFAR-100; 0.38 for Adience; and 0.12 for UTKFace. This is no surprise as $d_\theta(x)$ depends both on the location of the data points (which are fixed and immovable in a learning environment) and the choice of architectures/parameters.

Finally, we consider another partition of the datasets. Above, we consider the partition of the dataset which occurs by the class labels. With the Adience and UTKFace datasets, we have an additional partition by sensitive attributes. Adience admits partitions based off of gender; UTKFace admits partition by gender and ethnicity. We note that Adience and UTKFace use categorical labels for these multidimensional and socially complex concepts. We know this to be reductive and serves to minimize the contextualization within which race and gender derive their meaning (Hanna et al., 2020; Buolamwini and Gebru, 2018). Further, we acknowledge the systems and notions that were used to reify such data partitions and the subsequent implications and conclusions draw therefrom. We use these socially and systemically-laden partitions to demonstrate that the functions we define, $\widehat{I_P}$ and σ depend upon how the data are divided for analysis. To that end, the function $\widehat{I_P}$ is visualized in Figure 2.5. We observe that the Adience dataset, which exhibited some adversarial robustness bias in the partition on $\mathcal{C}$ only exhibits minor adversarial robustness bias in the partition on $\mathcal{S}$ for the attribute 'Female'. On the other hand, UTKFace which had signifiant adversarial robustness bias does exhibit the phenomenon for the sensitive attribute 'Black' but not for the sensitive attribute 'Female'.

This emphasizes that adversarial robustness bias is dependant upon the dataset and the partition. We will demonstrate later that it is also dependant on the choice of classifier. First, we talk about ways to approximate $d_\theta(x)$ for more complicated models.

2.7 Evaluation of Robustness Bias using Adversarial Attacks

As described in Section 2.4.1, we argued that adversarial attacks can be used to obtain upper bounds on $d_\theta(x)$ which can then be used to measure robustness bias. In this section we audit some popularly used models on datasets mentioned in Section 2.5 for robustness bias as measured using the approximation given by adversarial attacks.

2.7.1 Evaluation of $\widehat{I_P^{DF}}$ and $\widehat{I_P^{CW}}$

To compare the estimate of $d_\theta(x)$ by DeepFool and CarliniWagner, we first look at the signedness of $\sigma(P)$, $\sigma^{DF}(P)$, and $\sigma^{CW}(P)$. For a given partition P, $\sigma(P)$ captures the disparity in robustness between points in P relative to points not in P (see Eq 2.3). Considering all 151 possible partitions (based on class labels and sensitive attributes, where available) for all five datasets, both CarliniWagner and DeepFool agree with the signedness of the direct computation 125 times, i.e., $\mathbb{1}_P\left[\text{sign}(\sigma(P)) = \text{sign}(\sigma^{DF}(P))\right] = 125 = \mathbb{1}_P\left[\text{sign}(\sigma(P)) = \text{sign}(\sigma^{CW}(P))\right]$.

Further, the mean difference between $\sigma(P)$ and $\sigma^{CW}(P)$ or $\sigma^{DF}(P)$, i.e., $(\sigma(P) - \sigma^{DF}(P))$, is 0.17 for DeepFool and 0.19 for CarliniWagner with variances of 0.07 and 0.06 respectively.

There is 83% agreement between the direct computation and the DeepFool and CarliniWagner estimates of $\widehat{I_P}$. This behavior provides evidence that adversarial attacks provide meaningful upper bounds on $d_\theta(x)$ in terms of the behavior of identifying instances of robustness bias.

2.7.2 Audit of Commonly Used Models

We now evaluate five commonly-used convolutional neural networks (CNNs): Alexnet, VGG, ResNet, DenseNet, and Squeezenet. We trained these networks using PyTorch with standard stochastic gradient descent. We achieve comparable performance to documented state of the art for these models on these datasets. After training each model on each dataset, we generated adversarial examples using both methods and computed $\sigma(P)$ for each possible partition of the dataset. An example of the results for the UTKFace dataset can be see in Figure 2.7.

With evidence from Section 2.7.1 that DeepFool and CarliniWagner can approximate the robustness bias behavior of direct computations of d_θ, we first ask if there are any major differences between the two methods. *If DeepFool exhibits adversarial robustness bias for a dataset and a model and a class, does CarliniWagner exhibit the same? and vice versa?* Since there are 5 different convolutional models, we have $151 \cdot 5 = 755$ different comparisons to make. Again, we first look at the signedness of $\sigma^{DF}(P)$ and $\sigma^{CW}(P)$ and we see that $\mathbb{1}_P\left[\text{sign}(\sigma^{DF}(P)) = \text{sign}(\sigma^{CW}(P))\right] = 708$. This means there is 94% agreement between DeepFool and CarliniWagner about the direction of the adversarial robustness bias.

To investigate if this behavior is exhibited earlier in the training cycle than at the final, fully-trained model, we compute $\sigma^{CW}(P)$ and $\sigma^{DF}(P)$ for the various models and datasets for trained models after 1 epoch and the middle epoch. For the first epoch, 637 of the 755 partitions were internally consistent, i.e., the signedness of σ was the same in the first and last epoch, and 621 were internally consistent. We see that at the middle epoch, 671 of the 755 partitions were internally consistent for DeepFool and 665 were internally consistent for CarliniWagner. Unsurprisingly, this implies that as the training progresses, so does the behavior of the adversarial robustness bias. However, it is surprising that much more than 80% of the final behavior is determined after the first epoch, and there is a slight increase in agreement by the middle epoch.

We note that, of course, adversarial robustness bias is not necessarily an intrinsic value of a dataset; it may be exhibited by some models and not by others. However, in our studies, we see that the UTKFace dataset partition on Race/Ethnicity does appear to be significantly prone to adversarial attacks given its comparatively low $\sigma^{DF}(P)$ and $\sigma^{CW}(P)$ values across all models.

2.8 Evaluation of Robustness Bias using Randomized Smoothing

In Section 2.4.2, we argued that randomized smoothing can be used to obtain lower bounds on $d_\theta(x)$ which can then be used to measure robustness bias. In this section we audit popular models on a variety of datasets (described in detail in Section 2.5) for robustness bias, as measured using the approximation given by randomzied smoothing.

2.8.1 Evaluation of $\widehat{I_P^{RS}}$

To assess whether the estimate of $d_\theta(x)$ by randomized smoothing is an appropriate measure of robustness bias, we compare the signedness of $\sigma(P)$ and $\sigma^{RS}(P)$. When $\sigma(P)$ has positive sign, higher magnitude indicates a higher robustness of members of partition P as compared to members not included in that partition P; similarly, when $\sigma(P)$ is negatively signed, higher magnitude corresponds to lesser robustness for those members of partition P (see Eq 2.3). We may interpret shared signedness of both $\sigma(P)$ (where $d_\theta(x)$ is deterministic) and $\sigma^{RS}(P)$ (where $d_\theta(x)$ is measured by randomized smoothing as described in Section 2.4.2) as positive support for the $\widehat{I_P^{RS}}$ measure.

Similar to Section 2.7.1, we consider all possible 151 partitions across CIFAR-10, CIFAR-100, CIFAR-100Super, UTKFace and Adience. For each of these partitions, we compare $\sigma^{RS}(P)$ to the corresponding $\sigma(P)$. We find that their sign agrees 101 times, *i.e.*, $\mathbb{1}_P\left[\text{sign}(\sigma(P)) = \text{sign}(\sigma^{RS}(P))\right] = 101$, thus giving a 66.9% agreement. Furthermore, the mean difference between $\sigma(P)$ and $\sigma^{RS}(P)$, *i.e.*, $(\sigma(P) - \sigma^{RS}(P))$ is 0.08 with a variance of 0.19.

This provides evidence that randomized smoothing can also provide a meaningful estimate on $d_\theta(x)$ in terms of measuring robustness bias.

2.8.2 Audit of Commonly Used Models

We now evaluate the same models and all the datasets for robustness bias as measured by randomized smoothing. Our comparison is analogous to the one performed in Section 2.7.2 using adversarial attacks. Figure 2.8 shows results for all models on the UTKFace dataset. Here we plot σ_P^{RS} for each partition of the dataset (on x-axis) and for each model (y-axis). A darker color in the heatmap indicates high robustness bias (darker red indicates that the partition is *less* robust than others, whereas a darker blue indicates that the partition is *more* robust). We can see that some partitions, for example, the partition based on class label “40-60” and the partition based on race “black” tend to be less robust in the final trained model, for all models (indicated by a red color across all models). Similarly there are partitions that are more robust, for example, the partition based on class “0-15” and race “asian” end up being robust across different models (indicated by a blue color). Figure 2.6 takes a closer look at the distribution of distances for the UTKFace dataset when partitioned by race, showing that for different models different races can be more or less robust. Figures 2.6, 2.7 and 2.8 (we see similar trends for CIFAR-10, CIFAR-100, CIFAR-100Super and Adience) lead us to the following key conclusions:

Dependence on data distribution. The presence of certain partitions that show similar robustness trends as discussed above (*e.g.*see final trained model in Figs 2.8 and 2.7, the partitions by class “0-15” and race “asian” are more robust, whereas the class “40-60” and race “black” are less robust across *all models*) point to some intrinsic property of the data distribution that results in that partition being more (or less) robust regardless of the type decision boundary. Thus we conclude that robustness bias may depend in part on the data distribution of various sub-populations.

Dependence on model. There are also certain partitions of the dataset (e.g., based on the classes “15-25” and “60+” as per Fig 2.8) that show varying levels of robustness across different models. Moreover, even partitions that have same sign of $\sigma^{RS}(P)$ across different models have very different values of $\sigma^{RS}(P)$. This is also evident from Fig 2.6 which shows that the distributions of $d_\theta(x)$ (as approximated by all our proposed methods) for different races can be very different for different models. Thus, we conclude that robustness bias is also dependent on the learned model.

Role of pre-training. We now explore the role of pre-training on our measures of robustness bias. Specifically, we pre-train five of the six models (Resnet, Alexnet, VGG, Densenet, and Squeezenet) on ImageNet and then fine-tune on UTKFace. We also train a UTK classifier from scratch on UTKFace. Figures 2.8 and 2.7 shows robustness bias scores after the first epoch and in the final, fully-trained model. At epoch 1, we mostly see no robustness bias (indicated by close-to-zero values of $\sigma^{RS}(P)$) for UTK Classifier. This is because the model has barely trained by that first epoch and predictions are roughly equivalent to random guesses. In contrast, the other five models already have pre-trained ImageNet weights, and hence we see certain robustness biases that already exist in the model, even after the first epoch of training. Thus, we conclude that pre-trained models bring in biases due to the distributions of the data on which they were pre-trained and the resulting learned decision boundary after pre-training. We additionally see that these biases can persist even after fine-tuning.

2.8.3 Comparison of Randomized Smoothing and Upper Bounds

We have now presented two ways of measuring robustness bias: via upper bounds and via randomized smoothing. Figures 2.10, 2.12, 2.14, and 2.16 show the measures of robustness bias as approximated by CarliniWagner (σ_P^{CW}) and DeepFool (σ_P^{DF}) across different partitions of all the datasets. While there are important distinctions between the two methods, it is worth comparing them. To do this, we compare the sign of the randomized smoothing method and the upper bounds as

$$\mathbb{1}_P \left[\text{sign}(\sigma^{RS}(P)) = \text{sign}(\sigma^{DF}(P))\right]$$

and

$$\mathbb{1}_P \left[\text{sign}(\sigma^{RS}(P)) = \text{sign}(\sigma^{CW}(P))\right].$$

We see that there is some evidence that the two methods agree. The Adience, UTKFace, and CIFAR-10 dataset have strong agreement (at or above 75%) between the randomized smoothing for both types of upper bounds (DeepFool and CariliniWagner), while the CIFAR-100 dataset has a much weaker agreement (above but closer to 50%) and CIFAR-100Super has an approximately 66% agreement.

It is important to point out that it is not entirely appropriate to perform a comparison in this way. Recall that the upper bounds provide estimates of d_θ using a trained model. However, the randomized smoothing method estimates d_θ not directly with the trained model — instead it first modifies (smooths) the model of interest and then performs an estimation. Since the upper bounds and randomized smoothing methods are so different in practice, there may be no truly appropriate way to compare the results therefrom. Therefore, too much credence should not be placed on the comparison of these two methods. Both methods indicate the existence of the robustness bias phenomenon and can be useful in distinct settings.

2.9 Reducing Robustness Bias through Regularization

Motivated by evidence (see Section 2.5) that robustness bias exists in a diverse set of real-world models and datasets, we will now show that the expression of robustness bias can be included in an optimization. We do so in a natural way: by formulating a regularization term that captures robustness bias.

Recall the traditional Empirical Risk Minimization objective, ERM $:= l_{cls}(f_\theta(X), Y)$ where l_{cls} is cross entropy loss. Now we wish to model our measure of fairness (see Section 2.3) and minimize for it alongside ERM. We first write the empiric estimate of $RB(P, \tau)$ as $\tilde{R}B(P, \tau)$.

Recall the traditional Empirical Risk Minimization objective, ERM $:= l_{cls}(f_\theta(X), Y)$, where l_{cls} is cross entropy loss. Now we wish to model our measure of fairness (§2.3) and minimize for it alongside ERM. To model our measure, we first evaluate the following cumulative distribution functions:

$$\mathbb{P}_{x \in \mathcal{D}}\{d_\theta(x) > \tau \mid x \in P, y = \hat{y}\} = \frac{1}{\sum\limits_{x \notin P} \mathbb{1}\{y = \hat{y}\}} \sum_{\substack{x \notin P \\ y = \hat{y}}} \mathbb{1}\{d_\theta(x) > \tau\}$$

$$\mathbb{P}_{x \in \mathcal{D}}\{d_\theta(x) > \tau \mid x \notin P, y = \hat{y}\} = \frac{1}{\sum\limits_{x \in P} \mathbb{1}\{y = \hat{y}\}} \sum_{\substack{x \in P \\ y = \hat{y}}} \mathbb{1}\{d_\theta(x) > \tau\}$$

This gives us the empirical estimate of the robustness bias term $RB(P, \tau)$, parameterized by partition P and threshold τ, defined as Equation 2.7 below.

$$\tilde{RB}(P, \tau) = \Bigg| \frac{1}{\sum\limits_{x \notin P} \mathbb{1}\{y = \hat{y}\}} \sum_{\substack{x \notin P \\ y = \hat{y}}} \mathbb{1}\{d_\theta(x) > \tau\} - \frac{1}{\sum\limits_{x \in P} \mathbb{1}\{y = \hat{y}\}} \sum_{\substack{x \in P \\ y = \hat{y}}} \mathbb{1}\{d_\theta(x) > \tau\} \Bigg| \tag{2.7}$$

Now, for $\tilde{RB}$ as defined in Equation 2.7 to be computed and used, for example, during training, we must approximate a closed form expression of d_θ. To formulate this, we take inspiration from the way adversarial inputs are created using DeepFool (Moosavi-Dezfooli et al., 2016). Just like in DeepFool, we also approximate distance from f_θ considering f_θ to be linear (even though it may be a highly non-linear Deep Neural Network). Thus, we get,

$$d_\theta(x) = \left| \frac{f_\theta(x)}{||\nabla_x f_\theta(x)||} \right|. \tag{2.8}$$

By combining Equations 2.7 and 2.8, we recover $\tilde{RB}$, a computable estimate of the robustness bias term RB, as follows.

$$\tilde{RB}(P,\tau) = \Big| \frac{1}{\sum_{x \notin P} \mathbb{1}\{y=\hat{y}\}} \sum_{\substack{x \notin P \\ y=\hat{y}}} \mathbb{1}\{| \frac{f_\theta(x)}{||\nabla_x f_\theta(x)||} | > \tau\} - \frac{1}{\sum_{x \in P} \mathbb{1}\{y=\hat{y}\}} \sum_{\substack{x \in P \\ y=\hat{y}}} \mathbb{1}\{| \frac{f_\theta(x)}{||\nabla_x f_\theta(x)||} | > \tau\} \Big| \tag{2.9}$$

Finally, given scalar α, we minimize for the new objective function, *AdvERM*, as follows:

$$AdvERM := l_{cls}(f_\theta(X), Y) + \alpha \cdot \tilde{RB}(P, \tau).$$

2.9.1 Experimental Results using Regularized Models

Using our regularized objective, ADVERM, we re-train the model considering α and τ as hyperparameters, in addition to the traditional hyperparameters such as learning rate, momentum etc. Here, we evaluate the regularized model, after training for a number of values of α and τ, for CIFAR10 considering each class as the partition $\mathcal{P}$, and UTKFace with the sensitive attribute race as a partition $\mathcal{P}$. Figure 2.9 shows example results for the regularized models for the class “truck” in CIFAR10 and for the class “black” in UTKFace.

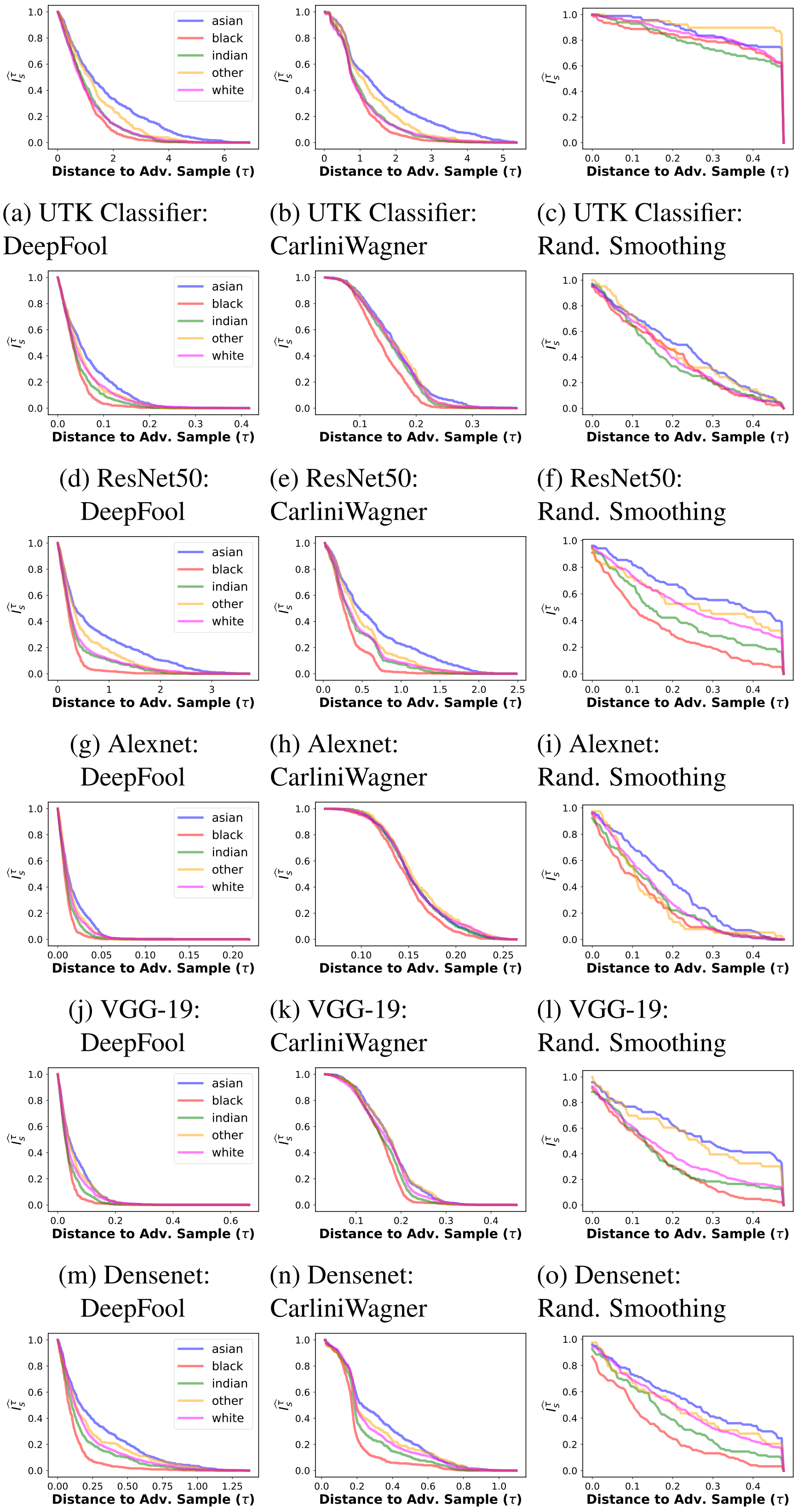
Figures 2.18, 2.19, 2.20, 2.21, 2.22, 2.23 shows how models trained with our proposed regularization term show lesser robustness bias. Figures 2.18, 2.19, and 2.20 correspond to CIFAR-10, while Figures 2.21, 2.22, and 2.23 correspond to UTKFace. For example, for Deep CNN trained on CIFAR-10, for the partition “cat”, we see that the distribution of distances become less disparate for the regularized model (Figure 2.18h) as compared to the original non-regularized model (Figure 2.18g). This trend persists across models and datasets. We provide a PyTorch implementation of the proposed regularization term in our accompanying code.

Across the two datasets, we see that for an appropriate α and τ, we are able to reduce the robustness disparity—*i.e.*, difference between blue and red curves—that existed in the original model. For these two datasets, this does not come at any cost of accuracy. We observe test set accuracy of 86.97% on CIFAR10 without regularization and 87.82% with regularization. Similarly for UTKFace we see accuracies of 66.25% and 65.52% for no regularization and with regularization respectively. We interpret our experiments as an indication that an optimization-based approach can play a part in a larger robustness bias-mitigation strategy, rather than serving as panacea.

2.10 Discussion and Conclusion

We propose a unique definition of fairness which requires all partitions of a population to be equally robust to minute (often adversarial) perturbations, and give experimental evidence that this phenomenon can exist in some commonly-used models trained on real-world datasets. Using these observations, we argue that this can result in a potentially unfair circumstance where, in the presence of an adversary, a certain partition might be more susceptible (*i.e.*, less secure). Susceptibility is prone to known issues with adversarial robustness such as sensitivity to hyperparameters (Tramer et al., 2020). Thus, we call for extra caution while deploying deep neural nets in the real world since this form of unfairness might go unchecked when auditing for notions that are based on just the model outputs and ground truth labels. We then show that this form of bias can be mitigated to some extent by using a regularizer that minimizes our proposed measure of robustness bias. However, we do not claim to "solve" unfairness; rather, we view analytical approaches to bias detection and optimization-based approaches to bias mitigation as potential pieces in a much larger, multidisciplinary approach to addressing these issues in fielded systems.

Indeed, we view our work as largely observational—we *observe* that, on many commonly-used models trained on many commonly-used datasets, a particular notion of bias, *robustness bias*, exists. We show that some partitions of data are more susceptible to two state-of-the-art and commonly-used adversarial attacks. This knowledge could be used for *attack* or to design *defenses*, both of which could have potential positive or negative societal impacts depending on the parties involved and the reasons for attacking and/or defending. We have also *defined* a notion of bias as well as a corresponding notion of fairness, and by doing that we admittedly toe a morally-laden line. Still, while we do use "fairness" as both a higher-level motivation and a lower-level quantitative tool, we have tried to remain ethically neutral in our presentation and have eschewed making normative judgements to the best of our ability.



(a) UTK Classifier: DeepFool

(b) UTK Classifier: CarliniWagner

(c) UTK Classifier: Rand. Smoothing

(d) ResNet50: DeepFool

(e) ResNet50: CarliniWagner

(f) ResNet50: Rand. Smoothing

(g) Alexnet: DeepFool

(h) Alexnet: CarliniWagner

(i) Alexnet: Rand. Smoothing

(j) VGG-19: DeepFool

(k) VGG-19: CarliniWagner

(l) VGG-19: Rand. Smoothing

(m) Densenet: DeepFool

(n) Densenet: CarliniWagner

(o) Densenet: Rand. Smoothing

(p) Squeezenet: DeepFool

(q) Squeezenet: CarliniWagner

(r) Squeezenet: Rand. Smoothing

Figure 2.6: UTKFace partitioned by race. We can see that across models, that different populations are at different levels of robustness as calculated by different proxies (DeepFool on the left, CarliniWagner in the middle and Randomized Smoothing on the right). This suggests that robustness bias is an important criterion to consider when auditing models for fairness.

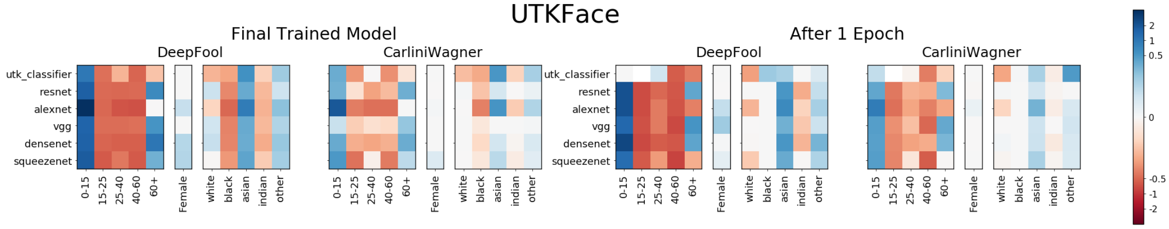


Figure 2.7: Depiction of σ_P^{DF} and σ_P^{CW} for the UTKFace dataset with partitions corresponding to the (1) class labels $\mathcal{C}$ and the, (2) gender, and (3) race/ethnicity. These values are reported for all five convolutional models both at the beginning of their training (after one epoch) and at the end. We observe that, largely, the signedness of the functions are consistent between the five models and also across the training cycle.

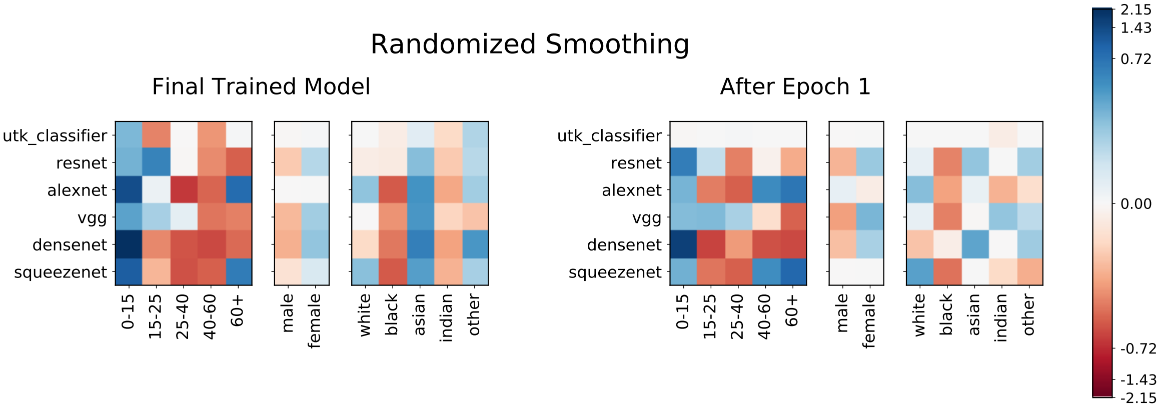
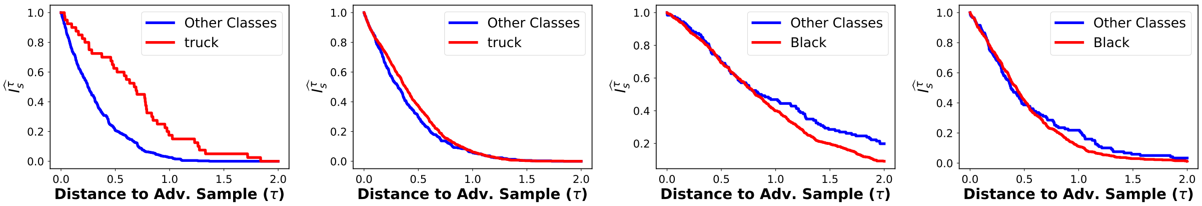


Figure 2.8: Depiction of σ_P^{RS} for the UTKFace dataset with partitions corresponding to the (1) class labels $\mathcal{C}$ and the, (2) gender, and (3) race/ethnicity. A more negative value indicates less robustness bias for the partition. Darker regions indicate high robustness bias. We observe that the trend is largely consistent amongst models and also similar to the trend observed when using adversarial attacks to measure robustness bias (see Figure 2.7).



(a) CIFAR10 (no reg.) (b) $\alpha = 0.1, \tau = 5.0.$ (c) UTKFace (no reg.) (d) $\alpha = 1.0, \tau = 5.0.$

Figure 2.9: In the unregularized model, "truck" in CIFAR10 tends to be more robust than other classes (2.9a); however, using AdvERM reduces that disparity (2.9b). We see similar behavior for UTKFace (2.6 & 2.9d).

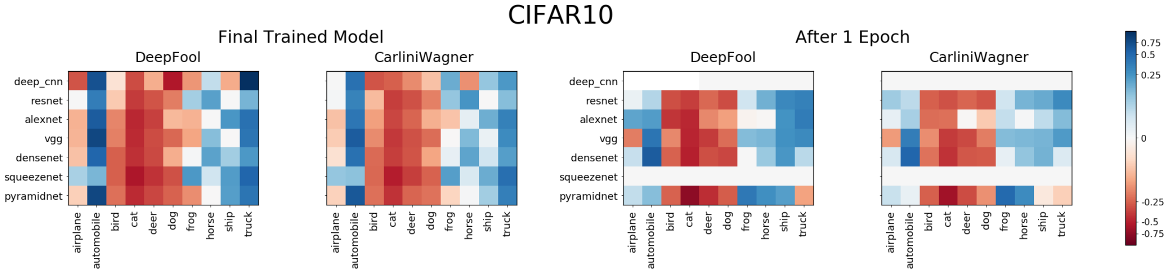


Figure 2.10: Depiction of σ_P^{DF} and σ_P^{CW} for the CIFAR10 dataset with partitions corresponding to the class labels $\mathcal{C}$. These values are reported for all five convolutional models both at the beginning of their training (after one epoch) and at the end. We observe that, largely, the signedness of the functions are consistent between the five models and also across the training cycle.

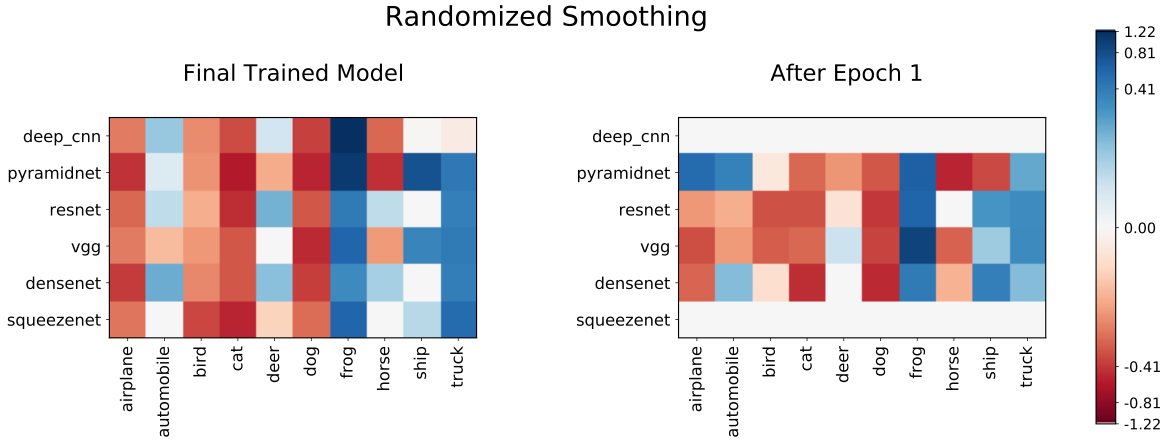


Figure 2.11: Depiction of σ_P^{RS} for the CIFAR10 dataset with partitions corresponding to the class labels $\mathcal{C}$.

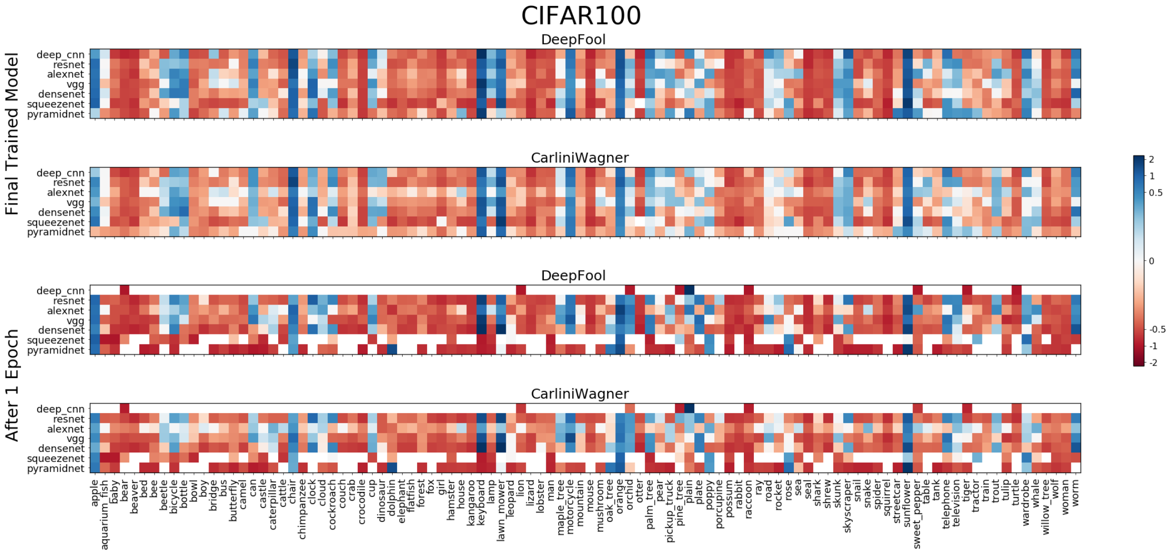


Figure 2.12: Depiction of σ_P^{DF} and σ_P^{CW} for the CIFAR100 dataset with partitions corresponding to the class labels $\mathcal{C}$. These values are reported for all five convolutional models both at the beginning of their training (after one epoch) and at the end. We observe that, largely, the signedness of the functions are consistent between the five models and also across the training cycle.

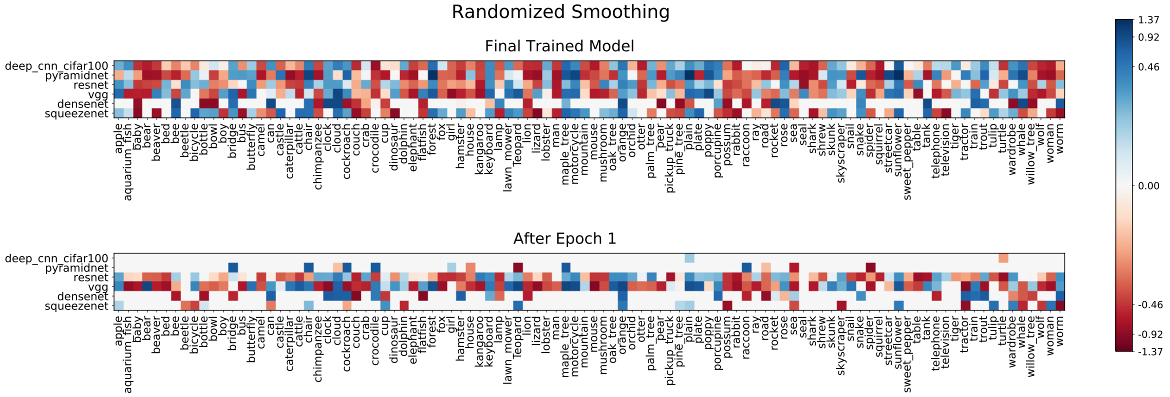


Figure 2.13: Depiction of σ_P^{RS} for the CIFAR100 dataset with partitions corresponding to the class labels $\mathcal{C}$.

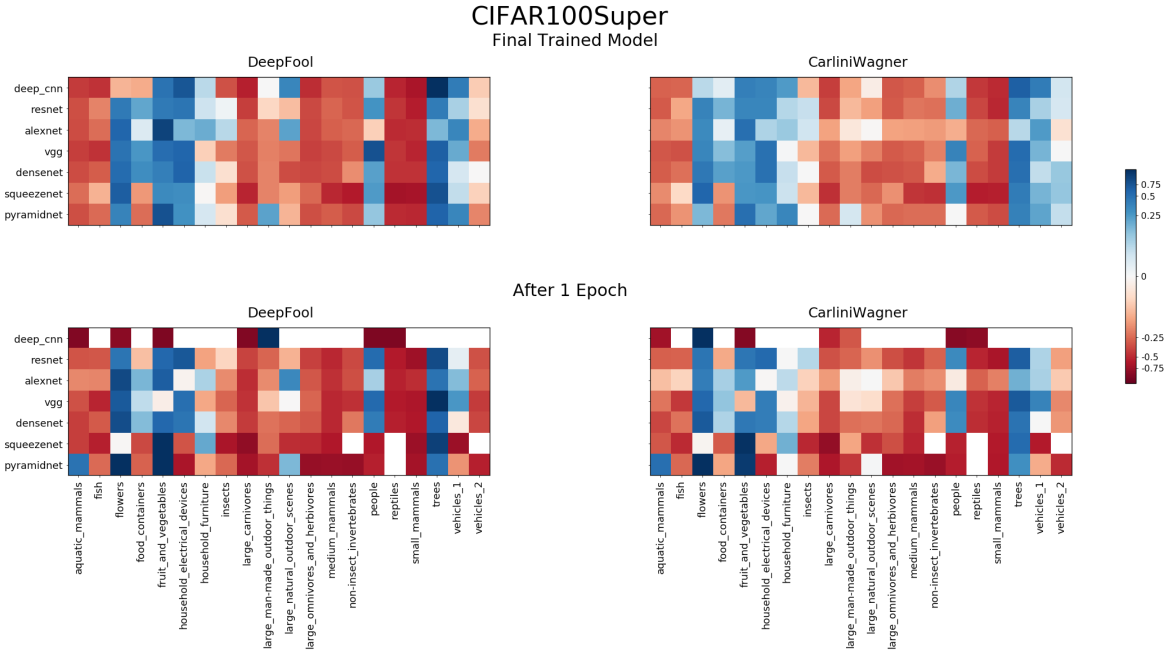


Figure 2.14: Depiction of σ_P^{DF} and σ_P^{CW} for the CIFAR100super dataset with partitions corresponding to the class labels $\mathcal{C}$. These values are reported for all five convolutional models both at the beginning of their training (after one epoch) and at the end. We observe that, largely, the signedness of the functions are consistent between the five models and also across the training cycle.

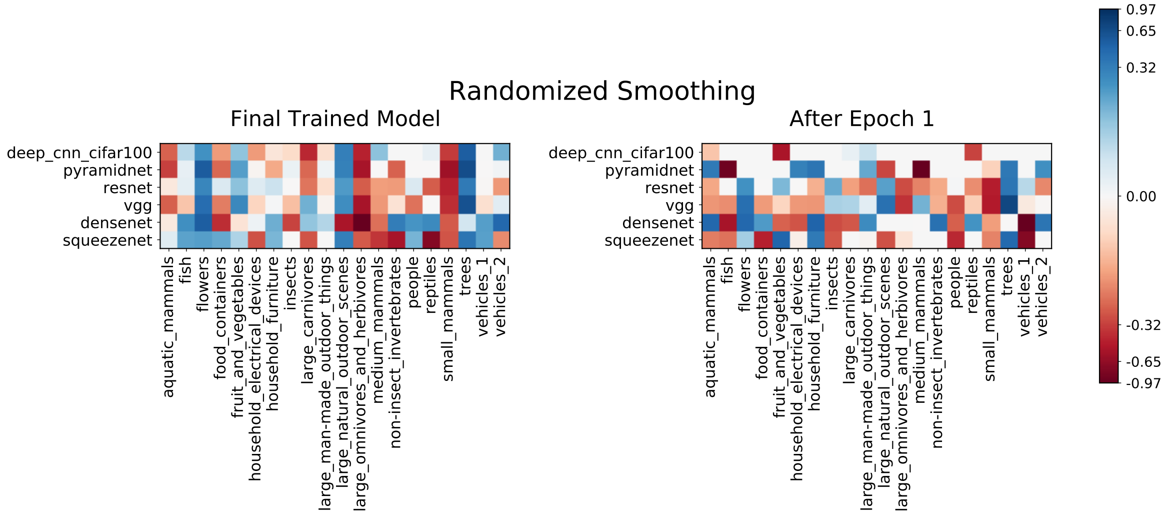


Figure 2.15: Depiction of σ_P^{RS} for the CIFAR100super dataset with partitions corresponding to the class labels $\mathcal{C}$.

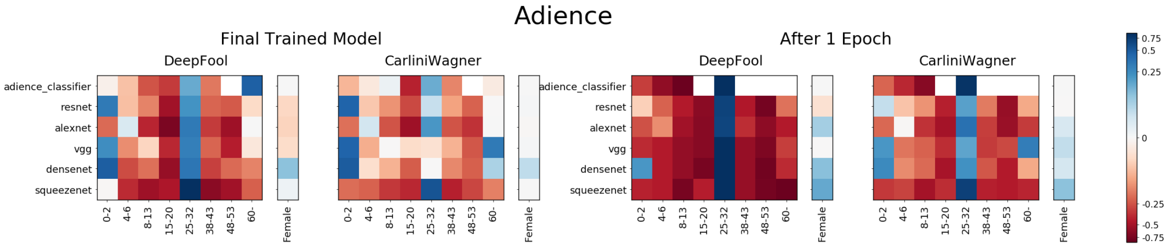


Figure 2.16: Depiction of σ_P^{DF} and σ_P^{CW} for the Adience dataset with partitions corresponding to the (1) class labels $\mathcal{C}$ and the and (2) gender. These values are reported for all five convolutional models both at the beginning of their training (after one epoch) and at the end. We observe that, largely, the signedness of the functions are consistent between the five models and also across the training cycle.

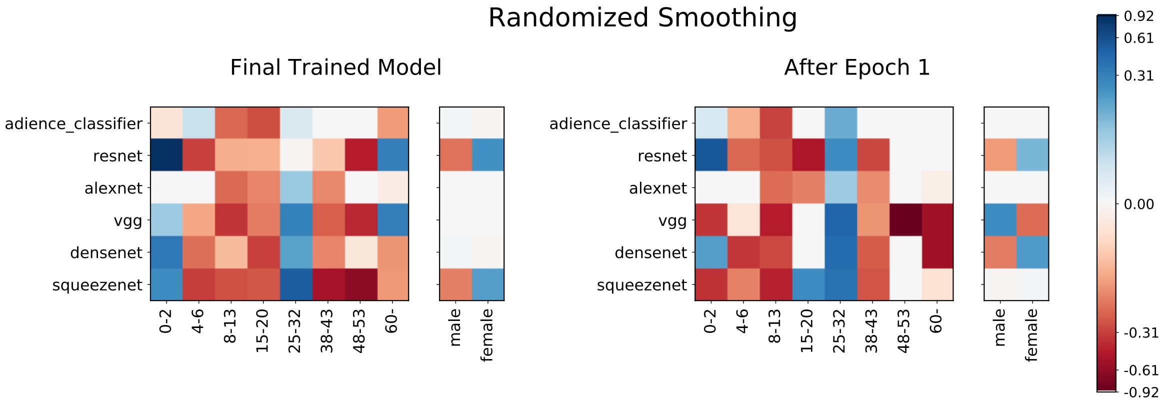
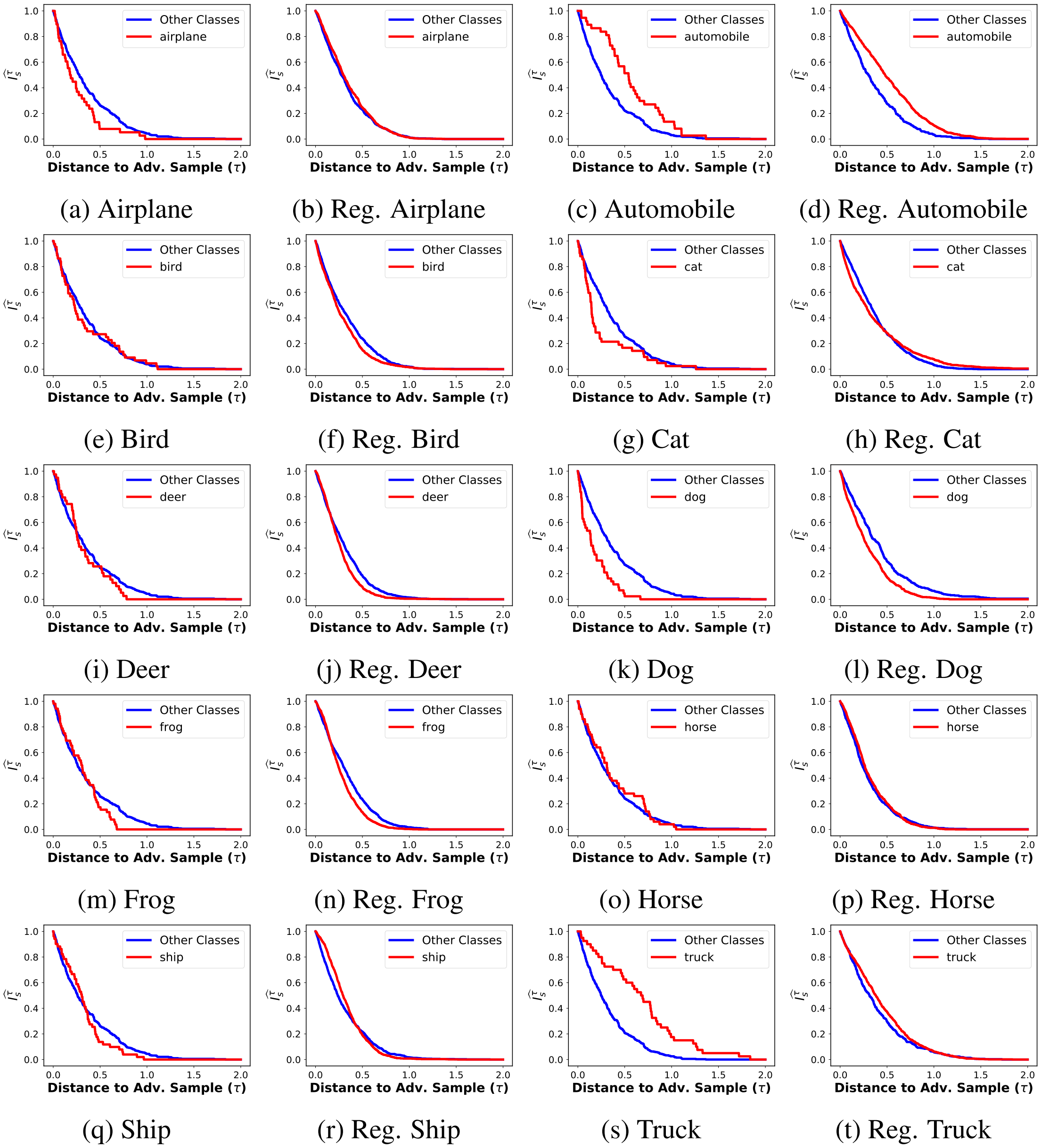


Figure 2.17: Depiction of σ_P^{RS} for the Adience dataset with partitions corresponding to the (1) class labels $\mathcal{C}$ and the and (2) gender.



(a) Airplane (b) Reg. Airplane (c) Automobile (d) Reg. Automobile

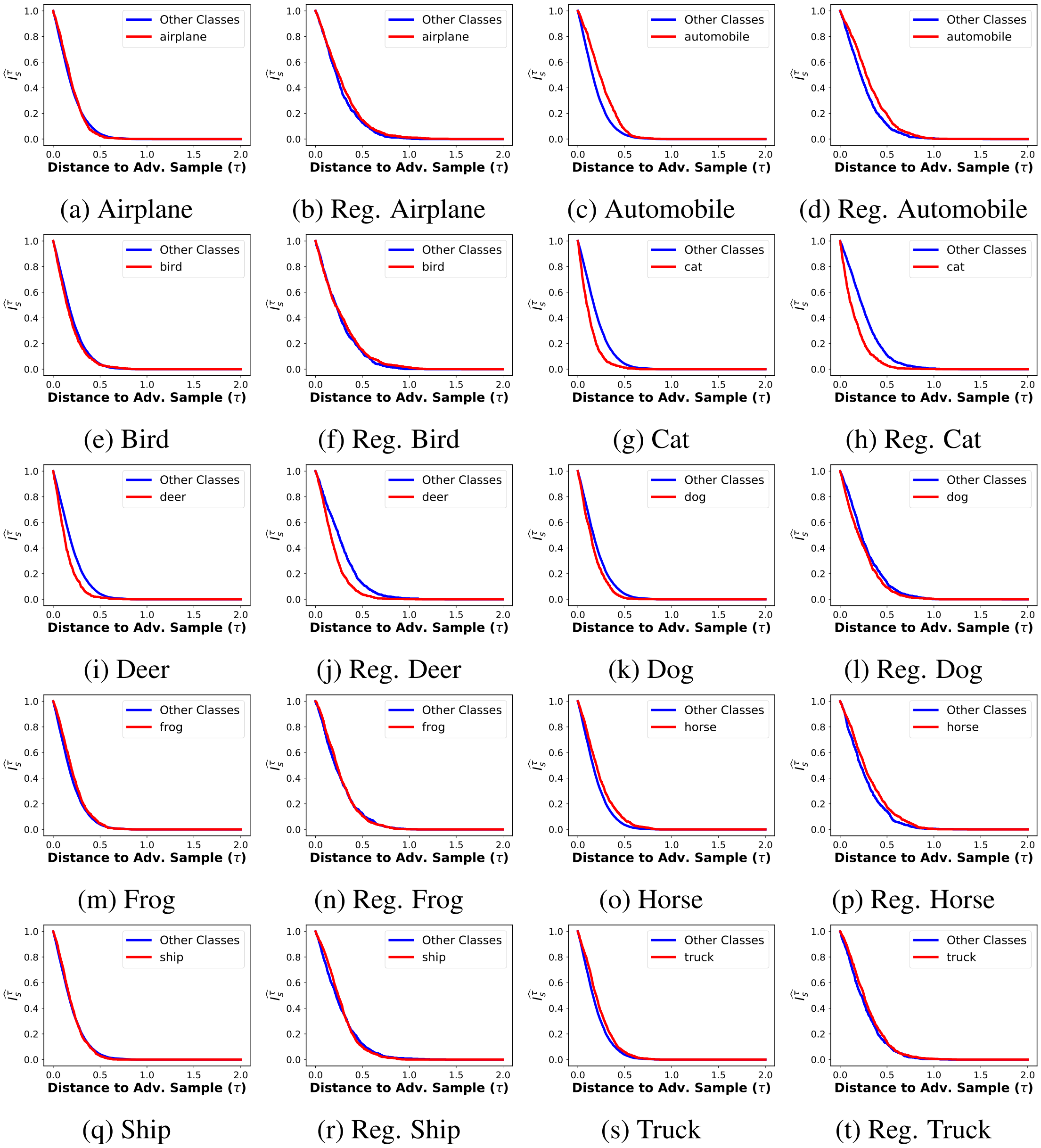
(e) Bird (f) Reg. Bird (g) Cat (h) Reg. Cat

(i) Deer (j) Reg. Deer (k) Dog (l) Reg. Dog

(m) Frog (n) Reg. Frog (o) Horse (p) Reg. Horse

(q) Ship (r) Reg. Ship (s) Truck (t) Reg. Truck

Figure 2.18: [Regularization] CIFAR10 - Deep CNN



(a) Airplane (b) Reg. Airplane (c) Automobile (d) Reg. Automobile

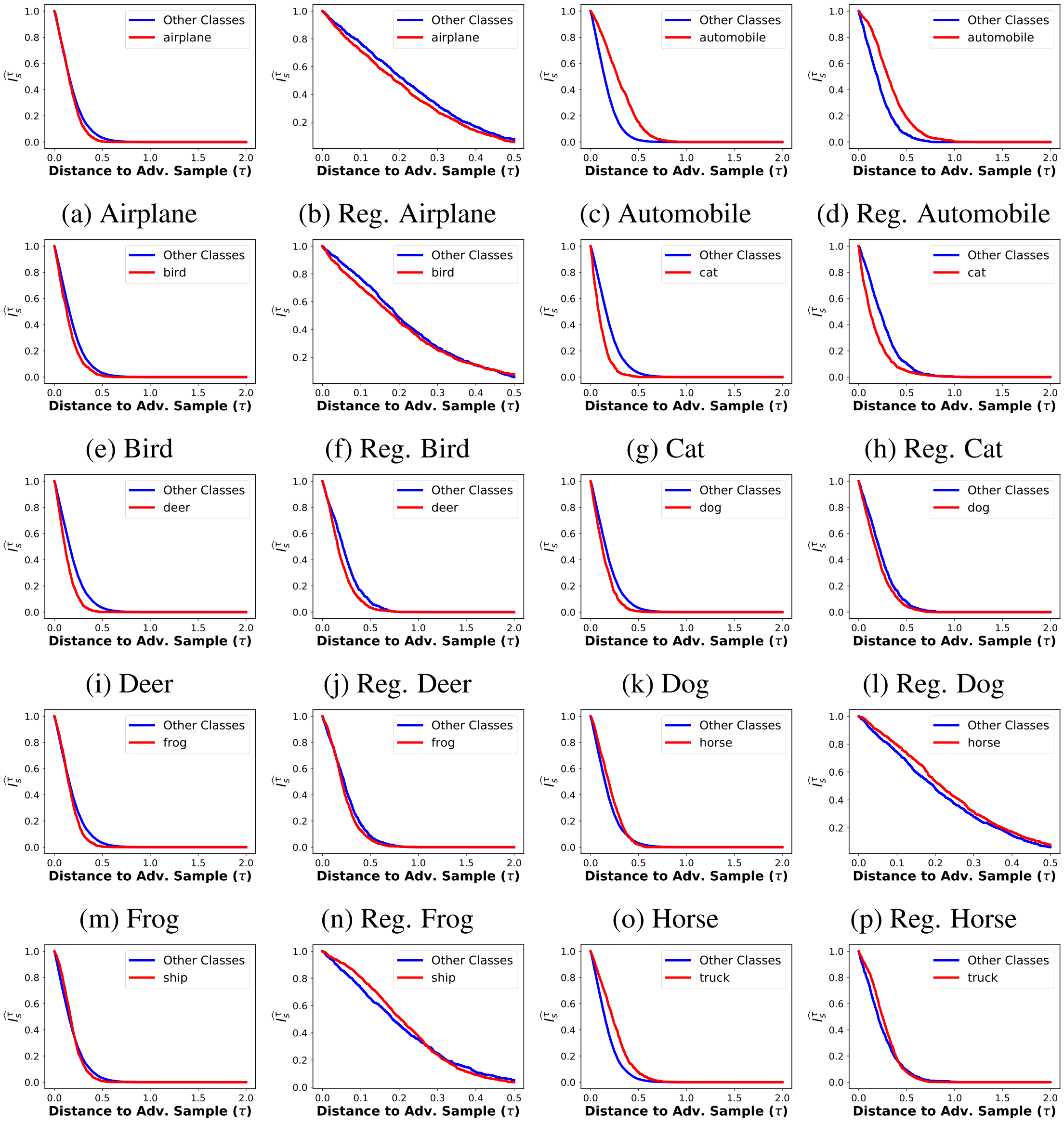
(e) Bird (f) Reg. Bird (g) Cat (h) Reg. Cat

(i) Deer (j) Reg. Deer (k) Dog (l) Reg. Dog

(m) Frog (n) Reg. Frog (o) Horse (p) Reg. Horse

(q) Ship (r) Reg. Ship (s) Truck (t) Reg. Truck

Figure 2.19: [Regularization] CIFAR10 - Resnet50



(a) Airplane

(b) Reg. Airplane

(c) Automobile

(d) Reg. Automobile

(e) Bird

(f) Reg. Bird

(g) Cat

(h) Reg. Cat

(i) Deer

(j) Reg. Deer

(k) Dog

(l) Reg. Dog

(m) Frog

(n) Reg. Frog

(o) Horse

(p) Reg. Horse

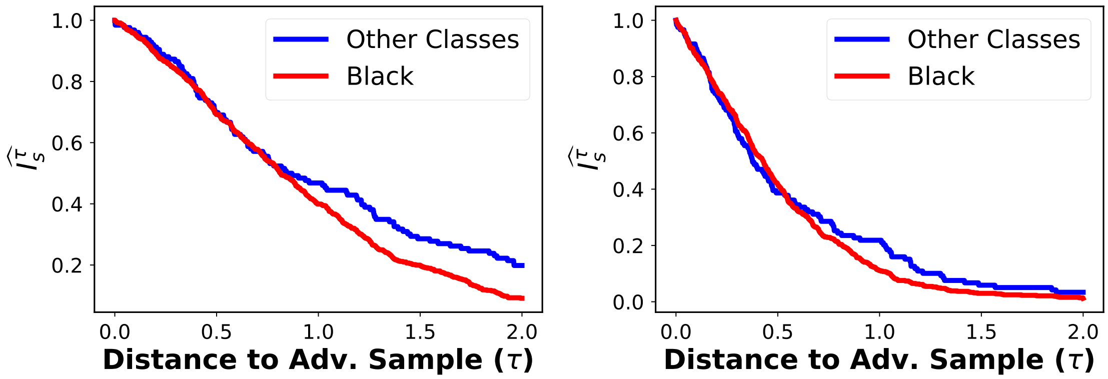
(q) Ship

(r) Reg. Ship

(s) Truck

(t) Reg. Truck

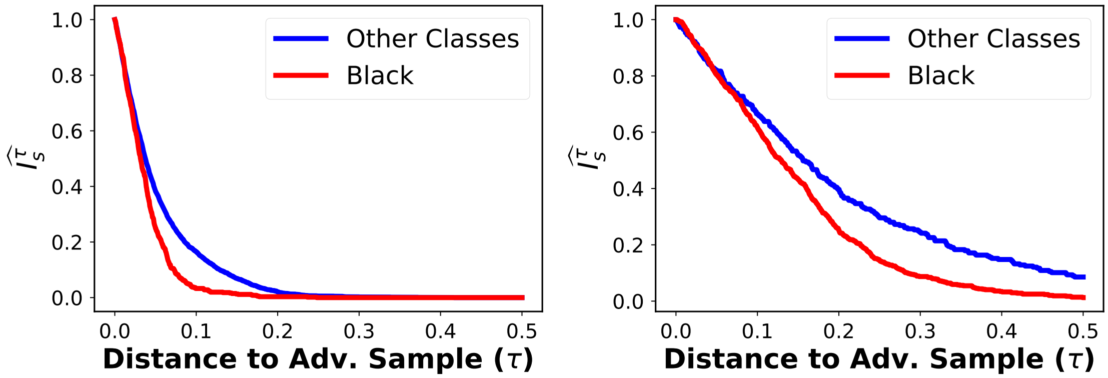
Figure 2.20: [Regularization] CIFAR10 - VGG19



(a) Partitioned by Race

(b) Regularized for Race

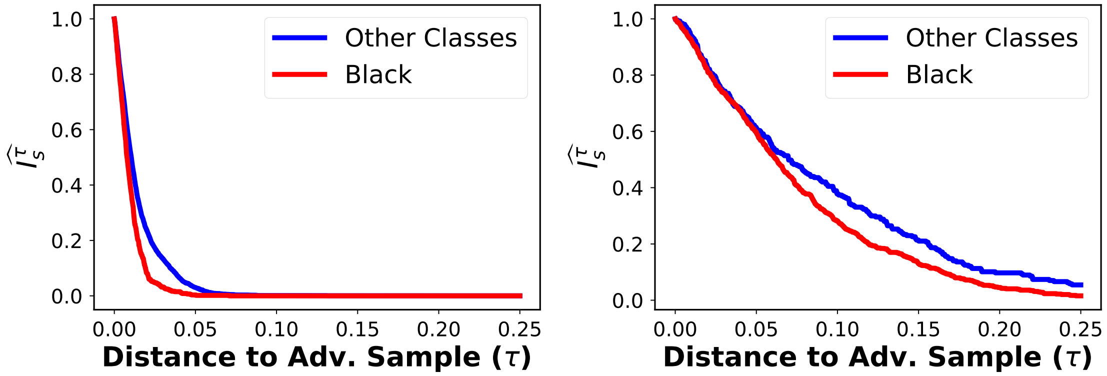
Figure 2.21: [Regularization] UTKFace partitioned by race - UTK Classifier.



(a) Partitioned by Race

(b) Regularized for Race

Figure 2.22: [Regularization] UTKFace partitioned by race - Resnet50.



(a) Partitioned by Race

(b) Regularized for Race

Figure 2.23: [Regularization] UTKFace partitioned by race - VGG.

Only a real one could tame me

Only the radio could play me

Oh, now you wish I was complacent

Boy, you musta mixed up our faces

HEATED

Beyoncé

Chapter 3: Robustness Disparities in Face Detection

This work was done in collaboration with George Z. Wei, Tom Goldstein and John P. Dickerson, and was presented at NeurIPS, 2022 (Dooley et al., 2022b).

Facial detection and analysis systems have been deployed by large companies and critiqued by scholars and activists for the past decade. Critiques that focus on system performance analyze disparity of the system's output, i.e., how frequently is a face detected for different Fitzpatrick skin types or perceived genders. However, we focus on the robustness of these system outputs under noisy natural perturbations. We present the first of its kind detailed benchmark of the robustness of three such systems: Amazon Rekognition, Microsoft Azure, and Google Cloud Platform. We use both standard and recently released academic facial datasets to quantitatively analyze trends in robustness for each. Across all the datasets and systems, we generally find that photos of individuals who are *older*, *masculine presenting*, of *darker skin type*, or have *dim lighting* are more susceptible to errors than their counterparts in other identities.

3.1 Introduction

Face detection identifies the presence and location of faces in images and video. In this work, face detection, also called face localization, refers to the task of placing a rectangle around the location of all faces in an image. Automated face detection is a core component of myriad systems—including *face recognition technologies* (FRT), wherein a detected face is matched against a database of faces, typically for identification or verification purposes. FRT-based systems are widely deployed (Hartzog, 2020; Derringer, 2019; Weise and Singer, 2020). Automated face recognition enables capabilities ranging from the relatively morally neutral (e.g., searching for photos on a personal phone (Google, 2021a)) to morally laden (e.g., widespread citizen surveillance (Hartzog, 2020), or target identification in warzones (Marson and Forrest, 2021)). Legal and social norms regarding the usage of FRT are evolving (e.g., Grother et al., 2019). For example, in June 2021, the first county-wide ban on its use for policing (see, e.g., Garvie, 2016) went into effect in the US (Gutman, 2021). Some use cases for FRT will be deemed socially repugnant and thus be either legally or *de facto* banned from use; yet, it is likely that pervasive use of facial analysis will remain—albeit with more guardrails than today (Singer, 2018).

One such guardrail that has spurred positive, though insufficient, improvements and widespread attention is the use of benchmarks. For example, in late 2019, the US National Institute of Standards and Technology (NIST) adapted its venerable Face Recognition Vendor Test (FRVT) to explicitly include concerns for demographic effects (Grother et al., 2019), ensuring such concerns propagate into industry systems. Yet, differential treatment by FRT of groups has been known for at least a decade (e.g., Klare et al., 2012; El Khiyari and Wechsler, 2016), and more recent work spearheaded by Buolamwini and Gebru (2018) uncovers unequal performance at the phenotypic subgroup level. That latter work brought widespread public, and thus burgeoning regulatory, attention to bias in FRT (e.g., Lohr, 2018; Kantayya, 2020).

One yet unexplored benchmark examines the bias present in a model's robustness (e.g., to noise, or to different lighting conditions), both in aggregate and with respect to different dimensions of the population on which it will be used. Many detection and recognition systems are not built in house, instead adapting an existing academic model or by making use of commercial cloud-based "ML as a Service" (MLaaS) platforms offered by tech giants such as Amazon, Microsoft, Google, Megvii, etc. With this in mind, our **main contribution** is a wide *robustness benchmark* of six different face detection models, three commercial-grade face detection systems (accessed via Amazon's Rekognition, Microsoft's Azure, and Google Cloud Platform's face detection APIs) and three high-performing academic face detection models (MogFace, TinaFace, and YOLO5Face). For fifteen types of realistic noise, and five levels of severity per type of noise (Hendrycks and Dietterich, 2019), we test all models against images in each of four well-known datasets. Across these more than $5\,000\,000$ noisy images from four commonly used academic datasets: Adience (Eidinger et al., 2014), Casual Conversations Dataset (Hazirbas et al., 2021), MIAP (Schumann et al., 2021), and UTKFace (Zhang et al., 2017). Additionally, to allow further research, we make our raw data available for exploration here.[1]

By benchmarking both commercial and academic models, we can understand two important insights: (1) audit the use-case of a company which takes open-source models to build in-house

[1] This work combines two unpublished papers which we wrote previously: (Dooley et al., 2021b) and (Dooley et al., 2022c). This submission expands on those papers' ideas and enhances them with more rigorous analysis.

facial recognition models, and (2) adjudicate corporation's claims of caring about demographic biases in their products by measuring the extent to which their models are less biased than academic models which have no fairness considerations. As such, we endeavor to answer three research questions:

(**RQ1**)**:** How robust are commercial and academic face detection models to natural types of noise?

(**RQ2**)**:** Do face detection models have demographic disparities in their performance on natural noise robustness tasks?

(**RQ3**)**:** Are the robustness disparities exhibited by commercial models more or less than the robustness disparities exhibited by academic models?

To answer these questions, we are motivated to understand how natural perturbations change the **system output.** We statistically analyze the performance of three common commercial facial detection providers and three state-of-the-art academic face detection models, comparing their performance and demographic disparities by comparing the output of the system on an unperturbed image with the output on a perturbed version of that image. This is interesting because it isolates the impact of the noise on the system, independent of the performance of the system. Thus, it makes comparing across systems easier. Focusing on **output** instead of system **performance** better isolates the impact of the stimulus of interest – the noise.

We observe that (RQ1) the leading face detection models show varying degrees of robustness to natural noise, but generally perform poorly on this task. Further, we conclude that (RQ2) these models do have demographic disparities which are statistically significant, and show a bias against individuals who are older, present as masculine, are darker skinned, and are dimly lit. Additionally, we see that (RQ3) these biases align with the commercial models, but that commercial model generally do not have lower level of disparity than the academic models.

Overall, our results suggest that regardless of a commercial company's commitments to equal treatment of different demographic groups, there are still pernicious problems with their products which treat demographic groups differently. We see further evidence that face detection is less robust to noise on older and masculine presenting individuals, which calls for future efforts to address this systemic problem. While our work indicates that the commercial providers are no worse on this important and socially impactful task than academics, we would hope to see that the commitments made by commercial companies would have them dedicate their vast resources and access to do better than comparatively under-resourced academics and substantially improve upon the robustness of their widely-used systems.

3.2 Related Work

We briefly overview additional related work in the two core areas addressed by our benchmark: robustness to noise and demographic disparity in facial detection and recognition. That latter point overlaps heavily with the fairness in machine learning literature; for additional coverage of that broader ecosystem and discussion around bias in machine learning writ large, we direct the reader to survey works due to Chouldechova and Roth (2018) and Barocas et al. (2019).

Demographic effects in facial detection and recognition. The existence of differential performance of facial detection and recognition on groups and subgroups of populations has been explored in a variety of settings (Klare et al., 2012; O'Toole et al., 2012; Buolamwini and Gebru, 2018; Raji and Buolamwini, 2019; Grother et al., 2019; Jain and Parsheera, 2021). In this work, we focus on *measuring* the impact of noise on a classification task, like that of Wilber et al. (2016); indeed, a core focus of our benchmark is to *quantify* relative drops in performance conditioned on an input datapoint's membership in a particular group. We view our work as a *benchmark*, that is, it focuses on quantifying and measuring, decidedly not providing a new method to "fix" or otherwise mitigate issues of demographic inequity in a system. Toward that latter point, existing work on "fixing" unfair systems can be split into three (or, arguably, four (Savani et al., 2020)) focus areas: pre-, in-, and post-processing. Pre-processing work largely focuses on dataset curation and preprocessing (e.g., Feldman et al., 2015b; Ryu et al., 2018; Quadrianto et al., 2019; Wang and Deng, 2020). In-processing often constrains the ML training method or optimization algorithm itself (e.g., Zafar et al., 2017b,a, 2019; Donini et al., 2018; Goel et al., 2018; Padala and Gujar, 2020; Agarwal et al., 2018; Wang and Deng, 2020; Martinez et al., 2020; Diana et al., 2020; Lahoti et al., 2020), or focuses explicitly on so-called fair representation learning (e.g., Adeli et al., 2021; Dwork et al., 2012; Zemel et al., 2013; Edwards and Storkey, 2016; Madras et al., 2018; Beutel et al., 2017; Wang et al., 2019b). Post-processing techniques adjust decisioning at inference time to align with quantitative fairness definitions (e.g., Hardt et al., 2016; Wang et al., 2020b).

Robustness to noise. Quantifying, and improving, the robustness to noise of face detection and recognition systems is a decades-old research challenge. Indeed, mature challenges like NIST's Facial Recognition Vendor Test (FRVT) have tested for robustness since the early 2000s (Phillips et al., 2007). We direct the reader to a comprehensive introduction to an earlier robustness challenge due to NIST (Phillips et al., 2011); that work describes many of the specific challenges faced by face detection and recognition systems, often grouped into Pose, Illumination, and Expression (PIE). It is known that commercial systems still suffer from degradation due to noise (e.g., Hosseini et al., 2017); none of this work also addresses the intersection of noise with bias, as we do.

Recently, *adversarial* attacks have been proposed that successfully break commercial face recognition systems (Shan et al., 2020; Cherepanova et al., 2021); we note that our focus is on *natural* noise, as motivated by Hendrycks and Dietterich (2019) with their ImageNet-C benchmark. Literature at the intersection of adversarial robustness and fairness is nascent and does not address commercial platforms (e.g., Singh et al., 2020; Nanda et al., 2021). To our knowledge, our work is the first systematic benchmark for commercial face detection systems that addresses, comprehensively, noise and its differential impact on (sub)groups of the population.

Academic Face Detection Models. Since 2012, neural-network-based face detectors have become ubiquitous in both industry and academia due to their comparative advantage in model capacity over traditional methods. As such, we are only going to focus on the prevailing approaches in deep face detection. According to Minaee et al. (2021), there are five main categories of face detectors. *Cascade-CNN Based Models* generally use convolutional neural networks (CNNs) that operate at various resolutions to produce detections that are then repeatedly refined (or "cascaded") through non-maximum suppression and bounding box regression to ultimately output final face detections (Li et al., 2015). *R-CNN Based Models* utilize a region proposal network to predict face regions and landmarks and then verify that the candidate regions are faces or not with a Regional CNN (Girshick et al., 2014). *Single Shot Detector (SSD) Models* discretize the output space of bounding boxes over different aspect ratios as well as scales then use the confidence scores to reshape the default boxes to better contain the detected faces by using convolutional features from different layers, usually the higher level layers (Liu et al., 2016). *Feature Pyramid Network (FPN) Based Models* upsample the convolutional features of higher (semantically richer) layers, aggregates them with those calculated in the initial forward pass to create semantically rich features at all image scales, then detects faces with each of these features at each layer (Lin et al., 2017). *Transformers Based Models* use the Transformer (Vaswani et al., 2017) (or the Vision Transformer (Dosovitskiy et al., 2021)) as the backbone for face detection. The academic models evaluated in this chapter fall into the FPN or SSD based detector categories and were chosen because they were top performers of the popular WIDER FACE (Xiong et al., 2015; Yang et al., 2016) benchmark.

Our work is most closely related to that of Jaiswal et al. (2022), who look at *adversarial noise* and how that effects "gender detection", "age prediction", and "smile detection". Jaiswal et al. (2022) explicitly do not examine detection as defined by face localization, which is the topic of this study. Further, their facial analysis technologies generally are downstream processes from the facial detection/localization technology in this chapter. Additionally, Majumdar et al. (2021) provide a similar experimental design as our work though for face verification, and on a significantly smaller set of image distortions and test images. We refer the reader to Singh et al. (2022) and Drozdowski et al. (2020) for surveys on bias in facial processing and biometrics.

3.3 Benchmark Design

In this section, we outline the details of our benchmark by describing the data we used, the protocol or method we employed to answer out research questions, and the evaluation metric. We also describe how our benchmark can be used by other researchers, the limitations of our benchmark, and give an important social context for our study in facial analysis technology.

3.3.1 Datasets

This benchmark uses four datasets to evaluate the robustness of three commercial and three academic face detection models. The datasets are described below.

The Open Images Dataset V6 – Extended; More Inclusive Annotations for People (**MIAP**)

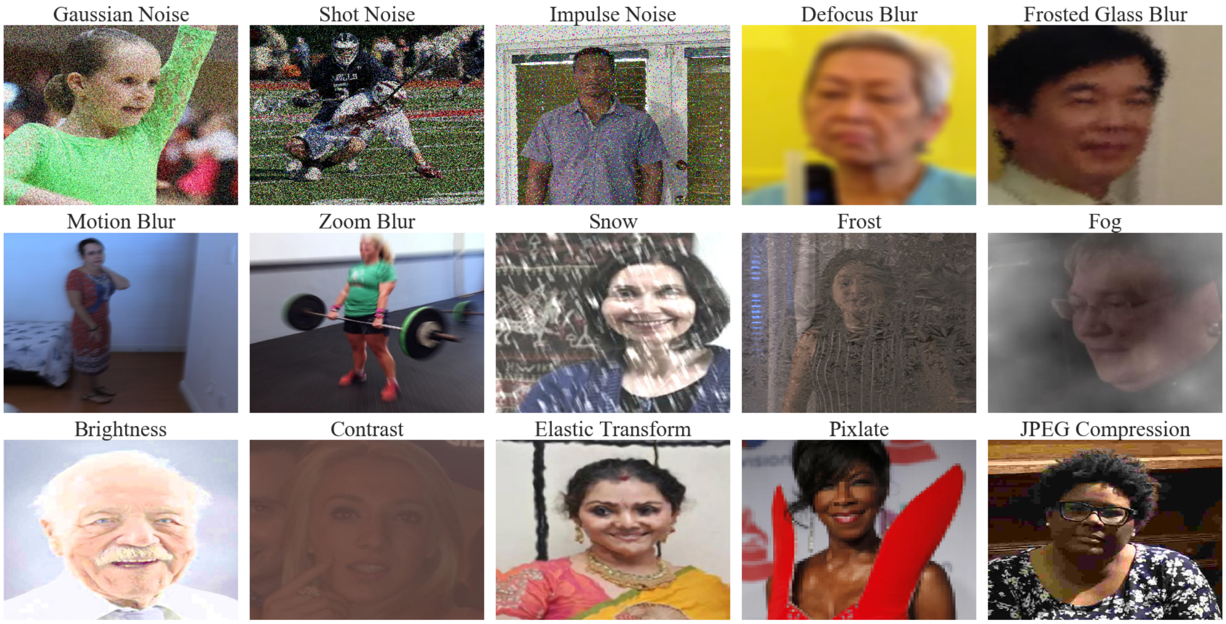


Figure 3.1: Our benchmark consists of 5,066,312 images of the 15 types of algorithmically generated corruptions produced by ImageNet-C. We use data from four datasets (Adience, CCD, MIAP, and UTKFace) and present examples of corruptions from each dataset here.

dataset (Schumann et al., 2021) was released by Google in May 2021 as a extension of the popular, permissive-licensed Open Images Dataset specifically designed to improve annotations of humans. For each image, every human is exhaustively annotated with bounding boxes for the entirety of their person visible in the image. Each annotation also has perceived gender (Feminine/Masculine/Unknown) presentation and perceived age (Young, Middle, Old, Unknown) presentation.

The Casual Conversations Dataset (**CCD**) (Hazirbas et al., 2021) was released by Facebook in April 2021 under limited license and includes videos of actors. Each actor consented to participate in an ML dataset and provided their self-identification of age and gender identity (coded as Female, Male, and Other), each actor's skin type was rated on the Fitzpatrick scale (Fitzpatrick, 1988), and each video was rated for its ambient light quality. For our benchmark, we extracted one frame from each video.

The **Adience** dataset (Eidinger et al., 2014) under a CC license, includes cropped images of faces from images "in the wild". Each cropped image contains only one primary, centered face, and each face is annotated by an external evaluator for age and perceived gender (Female/Male). The ages are reported as member of 8 age range buckets: 0-2; 3-7; 8-14; 15-24; 25-35; 36-45; 46-59; 60+.

Finally, the **UTKFace** dataset (Zhang et al., 2017) under a non-commercial license, contains images with one primary subject with annotated for age (continuous), perceived gender (Female/Male), and ethnicity (White/Black/Asian/Indian/Others) by an algorithm, then checked by human annotators.

For each of the datasets, we randomly selected a subset of images for our evaluation, with caps on the number of images from each intersectional identity equal to 1500. This reduces the effect of highly imbalanced datasets. We include a total of 66 662 clean images with 14 919 images from Adience; 21 444 images from CCD; 8194 images from MIAP; and 22 105 images form UTKFace.

For each dataset, we selected no more than 1500 images from any intersectional group. The final tallies of how many images from each group can be found in Tables 3.1, 3.2, 3.3, and 3.4.

3.3.1.1 Benchmark Protocol and Metrics.

Recall, our motivating question is how the noise impacts a model's *output*. To do this, each image was corrupted a total of 75 times, per the ImageNet-C protocol with the main 15 corruptions each with 5 severity levels. Examples of these corruptions can be seen in Figure 3.1. This resulted in a total of 5 066 312 images (including the original clean ones) which were each passed through

Table 3.1: Adience Dataset Counts

Age	Gender	Count
0-2	Female	684
	Male	716
3-7	Female	1232
	Male	925
8-14	Female	1353
	Male	933
15-24	Female	1047
	Male	742
25-35	Female	1500
	Male	1500
36-45	Female	1078
	Male	1412
46-59	Female	436
	Male	466
60+	Female	428
	Male	467

each of the six models.

Corruption information We evaluate 15 corruptions from Hendrycks and Dietterich (2019): Gaussian noise, shot noise, impulse noise, defocus blur, glass blur, motion blur, zoom blur, snow, frost, fog, brightness, contrast, elastic transforms, pixelation, and jpeg compressions. Each corruption is described in the Hendrycks and Dietterich (2019) paper as follows:

The first corruption type is Gaussian noise. This corruption can appear in low-lighting conditions. Shot noise, also called Poisson noise, is electronic noise caused by the discrete nature of light itself. Impulse noise is a color analogue of salt-and-pepper noise and can be caused by bit errors. Defocus blur occurs when an image is out of focus. Frosted Glass Blur appears with "frosted glass" windows or panels. Motion blur appears when a camera is moving quickly. Zoom blur occurs when a camera moves toward an object rapidly. Snow is a visually obstructive form of precipitation. Frost forms when lenses or windows are coated with ice crystals. Fog shrouds

Table 3.2: CCD Dataset Counts

Lighting	Gender	Skin	Age	Count
Bright	Female	Dark	19-45	1500
			45-64	1500
			65+	547
		Light	19-45	1500
			45-64	1500
			65+	653
	Male	Dark	19-45	1500
			45-64	1500
			65+	384
		Light	19-45	1500
			45-64	1500
			65+	695
	Other	Dark	19-45	368
			45-64	168
			65+	12
		Light	19-45	244
			45-64	49
Dim	Female	Dark	19-45	1500
			45-64	670
			65+	100
		Light	19-45	642
			45-64	314
			65+	131
	Male	Dark	19-45	1500
			45-64	387
			65+	48
		Light	19-45	485
			45-64	299
			65+	123
	Other	Dark	19-45	57
			45-64	26
			65+	3
		Light	19-45	27
			45-64	12

Table 3.3: MIAP Dataset Counts

AgePresentation	GenderPresentation	Count
Young	Unknown	1500
Middle	Predominantly Feminine	1500
	Predominantly Masculine	1500
	Unknown	561
Older	Predominantly Feminine	209
	Predominantly Masculine	748
	Unknown	24
Unknown	Predominantly Feminine	250
	Predominantly Masculine	402
	Unknown	1500

objects and is rendered with the diamond-square algorithm. Brightness varies with daylight intensity. Contrast can be high or low depending on lighting conditions and the photographed object's color. Elastic transformations stretch or contract small image regions. Pixelation occurs when upsampling a lowresolution image. JPEG is a lossy image compression format which introduces compression artifacts.

The specific parameters for each corruption can be found in the project's github at the corruptions file.

Benchmarks Costs Images were processed and stored within AWS's cloud using S3 and EC2. The experiments cost was \$17 507.55 and a breakdown can be found in Table 3.5.

API Parameters For the AWS DetectFaces API,[2] we selected to have all facial attributes returned. The Azure Face API[3] allows the user to select one of three detection models. We chose model `detection_03` as it was their most recently released model (February 2021) and was described

[2] https://docs.aws.amazon.com/rekognition/latest/dg/API_DetectFaces.html

[3] https://westus.dev.cognitive.microsoft.com/docs/services/563879b61984550e40cbbe8d/operations/563879b61984550f30395236

Table 3.4: UTKFace Dataset Counts

Age	Gender	Race	Count
0-18	Female	Asian	555
		Black	161
		Indian	350
		Others	338
		White	987
	Male	Asian	586
		Black	129
		Indian	277
		Others	189
		White	955
19-45	Female	Asian	1273
		Black	1500
		Indian	1203
		Others	575
		White	1500
	Male	Asian	730
		Black	1499
		Indian	1264
		Others	477
		White	1500
45-64	Female	Asian	39
		Black	206
		Indian	146
		Others	22
		White	802
	Male	Asian	180
		Black	401
		Indian	653
		Others	97
		White	1500
65+	Female	Asian	75
		Black	78
		Indian	43
		Others	10
		White	712
	Male	Asian	148
		Black	166
		Indian	91
		Others	5
		White	682

Table 3.5: Total Costs of Benchmark

Category	Cost
Azure Face Service	$4,270.58
AWS Rekognition	$4,270.66
Google Cloud Platform	$7,230.47
S3	$1,003.83
EC2	$475.77
Tax	$256.24
Total	$17,507.55

to have the highest performance on small, side, and blurry faces, since it aligns with our benchmark intention. This model does not return age or gender estimates (though model `detection_01` does).

These experiments were conducted in July 2021.

Evaluation Metrics To evaluate the change that image corruptions have to face detection systems, we measure the precision of the corrupted images while using the detections from the clean image as ground truth. While this approach obviates the need for real ground truth bounding boxes, it is also a principled measurement strategy for our main research question. Since we are primarily interested in how the system is affected by the corruption, this metric is superior to using real ground truth bounding boxes. This follows because we're interested in isolating the change in a system under a corruption which is exactly what this method measures.

To compute precision, we first observe the face detections on each clean image. After subsequently observing the face detection of a corrupted version of the clean image, we compute the image-level precision and recall for the corrupted image while using whatever the clean image's detections were as ground truth.

We evaluate the change of the face systems under perturbations using the standard object detection metric: mean average precision (mAP). We use the standard implementation of the mAP metric by COCO (Lin et al., 2014). Values reported below are mAP scores averaged over intersection over union (IoU) thresholds between 0.5 and 0.95 in intervals of 0.05. Below we call this metric Average Precision because we only have one class so the “mean" in mean average precision is trivial. Since we are interested in the system change under perturbation, and because none of the datasets have underlying ground truths, we treat the system output of the clean image as ground truth. A visual depiction of this process can be found in Figure 3.2.

We also investigate the significance of whether two groups are equally treated by a model under each metric by performing statistical tests. We observe bias by first performing a Kruskal-Wallis Rank Sum Test between explanatory and response variables which indicate whether two or more groups are treated equally or not. In the case where there is enough evidence to show that groups are treated differently, we then run the Pairwise Wilcoxon Rank Sum Tests to observe which groups have significantly different treatment and in which direction. All statistical tests are reported with $\alpha = 0.05$ with Bonferroni-Holm corrections. Each claim we make across datasets is done by looking at the trends in each dataset and are inherently qualitative.

We visually represent our results in Figures 3.4-3.7 by examining odds ratios between two categories of a sociodemographic variable across each model and dataset. For each pair of subgroups, like Middle-aged and Older subjects in Figure 3.5, we calculate the odds of each for each subgroup, $Odds_{middle}$ and $Odds_{older}$ and then look at their ratio: $Odds_{middle}/Odds_{older}$. When this value is greater than 1, like in Figure 3.5, it means the odds of higher performance are larger for middle aged group is higher than the older group. We conclude that there is a bias against older subjects. When the error bounds do not cross 1, this means that this disparity is

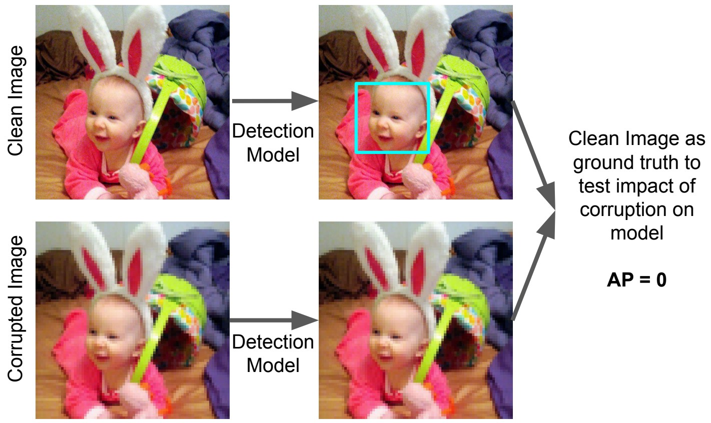

Clean Image
Corrupted Image
Detection Model
Detection Model
Clean Image as ground truth to test impact of corruption on model
AP = 0

Figure 3.2: Depiction of how Average Precision (AP) metric is calculated by using clean image as ground truth.

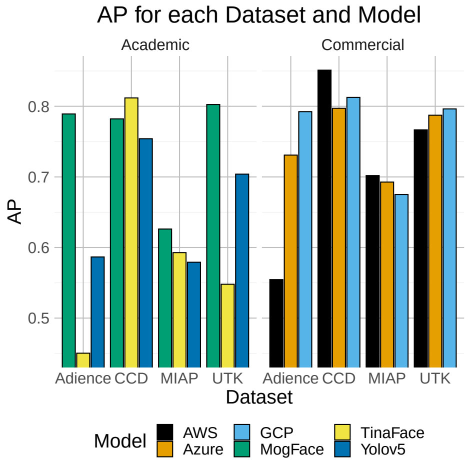


Figure 3.3: Overall performance (AP) of each model on each dataset.

statistically significant as well.

3.3.1.2 How to Use our Benchmark.

There are three main ways that our benchmark could be used by future researchers and practitioners. First, the analysis code, data, and results are being released publicly. New models that are built, either in academia or industry, can be easily benchmarked against our framework, and progress in this space can be tracked by the research community. Indeed, it is our intention to communicate our results to standards bodies such as NIST for inclusion in, or influence on, their long-running FRVT gauntlet. Second, the comparison across types of models (in our case, academic and commercial) could be adopted by more algorithmic audits. For example, in many areas (language models for text generation, diffusion models for text to image tasks, myriad object detection tasks) academic, industry-funded but open-sourced, and industry-funded and closed-source models compete across various metrics, and comparing and contrasting appropriately-defined bias metrics across those verticals is of practical importance. Third, well-founded and quantitative studies may be of use to policymakers. As discussed in

Section 3.1, facial analysis is a topic of great regulatory and legislative interest at this moment, and informing all sides—policymakers, the public, and providers of facial analysis technology—will lead to more clear and educated discussion and norm setting.

3.3.1.3 What is not included in this study.

There are three main things that this benchmark does not address. First, we do not examine cause and effect. We report inferential statistics without discussion of what generates them. Second, we only examine the types of algorithmicaly generated natural-like noise present in the 15 corruptions. We explicitly do not study or measure robustness to other types of changes to images, for instance adversarial noise, camera dimensions, etc. Finally, we do not investigate algorithmic training. We do not assume any knowledge of how the commercial system was developed or what training procedure or data were used.

3.3.1.4 Social Context.

This benchmark relies on socially constructed concepts of gender presentation and skin-tone/race and the related concept of age. While this benchmark analyzes phenotypal versions of these from metadata on ML datasets, it would be wrong to interpret our findings absent a social lens of what these demographic groups mean inside a society. We guide the reader to Benthall and Haynes (2019) and Hanna et al. (2020) for a look at these concepts for race in machine learning, and Hamidi et al. (2018) and Keyes (2018) for similar looks at gender.

3.4 Results

3.4.1 RQ1: Overall Model Performance

To answer RQ1 and to provide a baseline for comparison later in the analysis, we examine the overall performance of each model on each dataset, presented in Figure 3.3. We see from the outset that we can answer RQ1 affirmatively: face detection models sometimes struggle significantly with robustness to noise. Commercial models as a whole outperform the academic models on every dataset – however there are individual models in each category which break this conclusion. For example, the academic model MogFace performs significantly better than all the commerical models on UTKFace, though as a whole the academic models are inferior to the commercial ones.

Within in each class of model, commercial and academic, there is not a clear top model. However, we note that on the academic model side, MogFace significantly outperforms the other two models on every dataset except CCD. It is unknown as to why MogFace has such high performance, but we hypothesize a reason for what might explain this. MogFace was published very recently (late 2021), and perhaps much more recently than the commercial models. Only Azure indicates when its model was released (February 2021). The analysis of the commercial providers was also done prior to the release of MogFace. While more contemporary models do not necessarily imply better performance, this could be playing a role.

3.4.1.1 Performance of Individual Perturbations

Recall that there are four types of ImageNet-C corruptions: noise, blur, weather, and digital. On Adience, Brightness is the easiest corruption and noise is the hardest on five of the six models – GCP performs best on Pixelate and worst on Snow. On CCD, all models perform best on Glass Blur but worst on Zoom Blur.

Again, he zoom blur corruption proves particularly difficult on the MIAP datasets – it is the worst performer on all models for this dataset, whereas Brightness is the easest on four of the six models. On UTKFace, elastic-transform is a notable corruption which the models struggle with – all models except TinaFace and Yolo5Face perform worst on elastic tansform and UTKFace; all models except GCP perform best on Brightness. TinaFace and Yolo5Face struggle very significantly with the noise corruptions on UTKFace. Further details and analysis can be found in Table A.1 and Appendix A.1.2.

3.4.2 RQ2: Demographic Disparities in Noise Robustness

We now turn our attention to answer RQ2: do face detection models have demographic disparities in their performance on noise robustness tasks? Each dataset we analyze has both perceived gender and perceived age labels and CCD has perceived skin type and lighting conditions.

3.4.2.1 Gender Disparities

We begin by first pausing to note that the labels we have for perceived gender were in all cases provided by a third-party human reviewer, and the labels fall within the gender binary. The one exception is the MIAP dataset which reports a category of “Unknown” for times when the

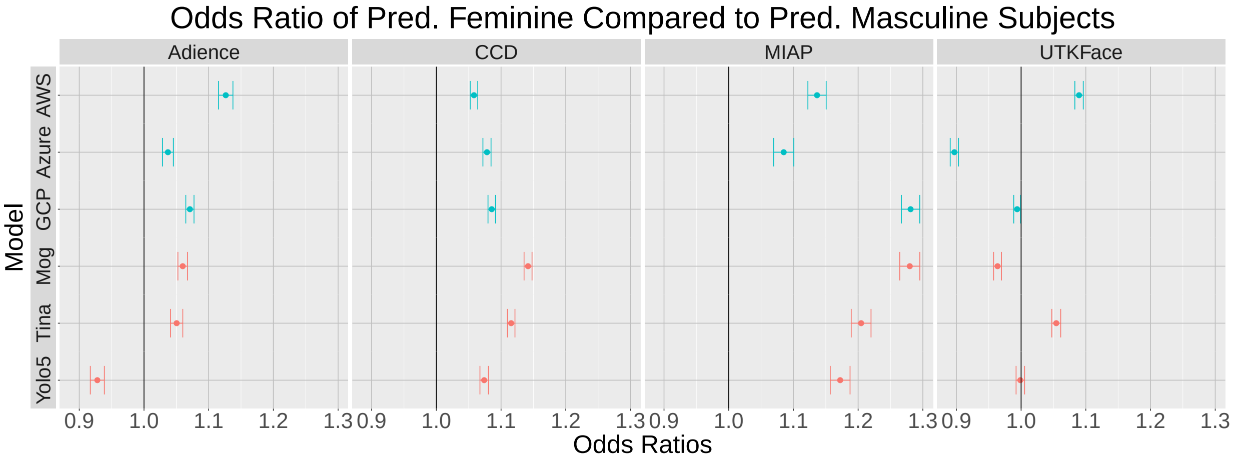


Figure 3.4: Gender disparity plots for each dataset and model. Values below 1 indicate that predominantly feminine presenting subjects are more susceptible to noise-induced changes. Values above 1 indicate that predominantly masculine presenting subjects are are more susceptible to noise-induced changes. Error bars indicate 95% confidence.

human reviewers were unable to reach a decision on the perceived gender of the subject. While gender is not binary and gender identity is not something which third party reviewers can infer, we use the perceived gender concept in our work to measure how model performance may differ for people who present gender differently.

We visually depict the performance of each model on each dataset in Figure 3.4 broken down by perceived gender. We analyze the observed perceived gender disparities for each dataset separately with a report of the odds ratio of feminine presenting individuals over masculine presenting individuals. Recall, values over 1 indicate higher performance on those whose are feminine presenting, and values less than 1 indicate higher performance on those who are masculine presenting.

We observe, qualitatively, across the 24 dataset and model combinations, there is a bias against masculine presenting individuals in 19 of them, there is a bias against feminine presenting subjects in 4, and there is no bias in one. This is a rather surprising result as previous reports indicate biases against feminine presenting individuals in facial recognition technology.

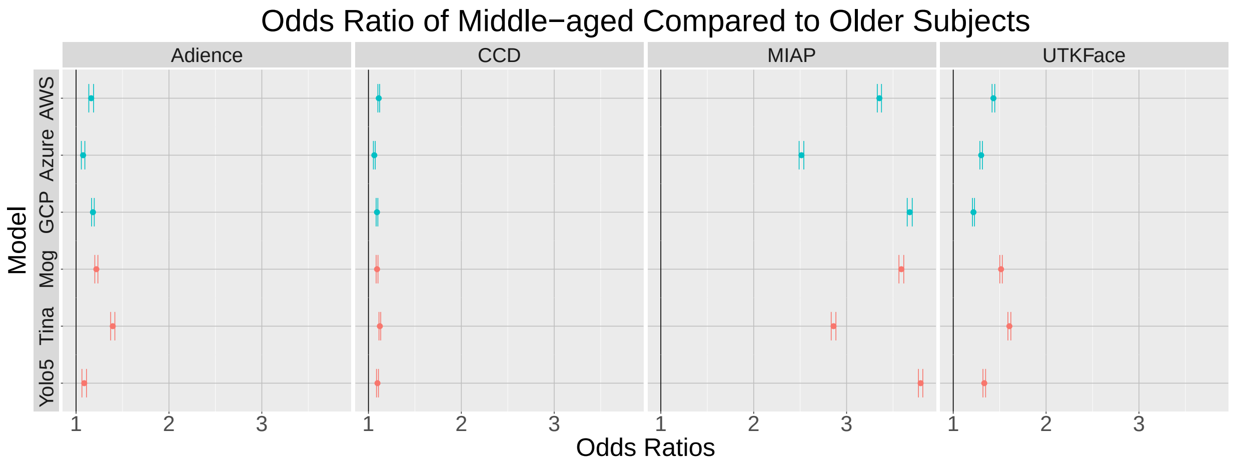


Figure 3.5: Age disparity plots for each dataset and model. Values greater than 1 indicate that older subjects are more susceptible to noise-induced changes compared to middle aged subjects. Error bars indicate 95% confidence.

We further observe that the UTKFace dataset has the lowest robustness bias for perceived gender across all the models. This indicates that the dataset itself is an important tool in the measurement of algorithmic disparities and suggests that future work in this domain area should greatly expand their studies to incorporate multiple datasets.

3.4.2.2 Age Disparities

We move on to a discussion of the age disparities present in these models and datasets. We report the results of this age disparity in Figure 3.5. We note again, that age labels are given by perceived age of the subject in the image. Adience provides disparate age categories, MIAP provides age groupings (Young, Middle, Older, and Unknown) and UTKFace natively provides a numeric value. Since numeric age values from UTKFace are likely misspecified as it is nearly impossible to correctly predict a person's age from a photo, we bin these numeric values into four buckets of (0-18), (19-45), (45-65) and (65+).

Qualitatively, looking at all these results, we observe that the oldest group always is more

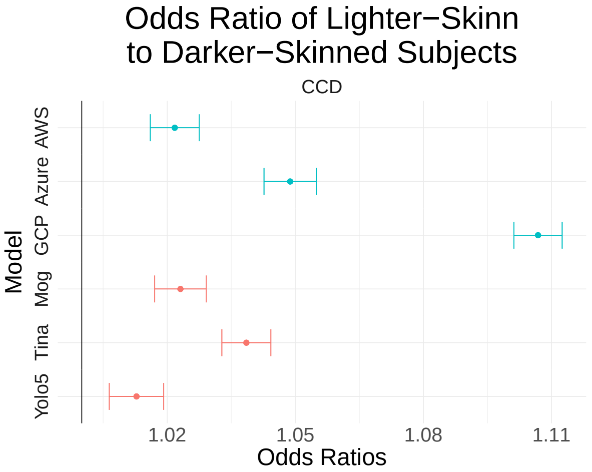


Figure 3.6: Skin type disparity plots for CCD. Values above 1 indicate that darker-skinned subjects are more susceptible to noise-induced changes. Error bars indicate 95% confidence.

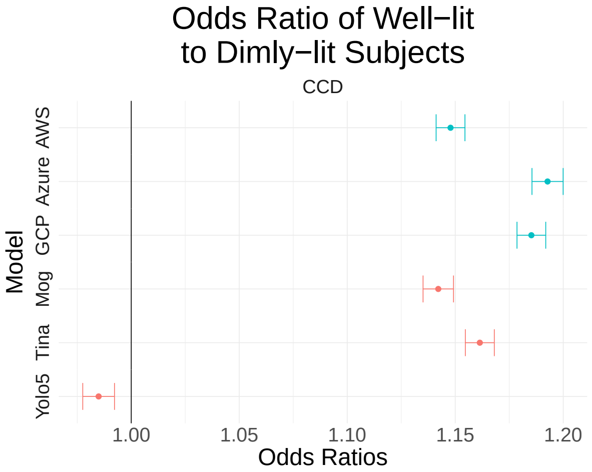


Figure 3.7: Lighting disparity plots for CCD. Values above 1 indicate that dimly-lit subjects are more susceptible to noise-induced changes. Error bars indicate 95% confidence.

susceptible to noise-induced changes compared to middle aged individuals. Quantiatively as well, we see that the oldest group is always statistically significantly the lowest performer of the groups. We note that while there may be differences in the sample sizes of these groups, the statistical tests are robust to these differences and account for sample size differences. Statistical test results for Pairwise Wilcoxon Rank Sum Tests can be found in the Appendix.

For MIAP, we observe significantly higher biases against older individuals than we do for the other datasets. We hypothesize that this might be due to the way in which the MIAP dataset was collected and the nature of the more natural images of entire scenes with sometimes multiple faces in them.

3.4.2.3 Skin Type and Lighting Disparities

The only dataset which includes metadata on skin type and illumination is the CCD dataset. As was customary at the time of the dataset release, CCD reports annotator provided Fitzpatrick skin type labels which we split the into two groups: Lighter (for ratings I-III) and Darker for ratings (IV-VI).

We observe a statistically significant bias against dark skinned individuals across every model, as can be seen in Figure 3.6. We further report that the bias between skin types is highest in the youngest groups; and this bias decreases in older groups. We also see a similar trend in the intersectional identities available in the CCD metadata (age, perceived gender, and skin type). We see that in every identity (except for 45-64 year old and Other gendered) the darker skin type has statistically significant lower AP. This difference is particularly stark in 19-45 year old, masculine subjects.

Lighting condition is also included as a metadata label in the CCD dataset. In Figure 3.7, we see that every model, except for YOLO5Face, exhibits behavior such that dimly lit images are more susceptible to noise-induced changes than brightly lit images. Interestingly and across the board, we generally see that the disparity in demographic groups decreases between bright and dimly lit environments. For example, the difference in precision between dark and light skinned subjects decreases to zero in dimly lit environments. This is also true for age groups. However, this is not true for individuals with gender presentations as Other or omitted.

3.4.3 RQ3: Disparity Comparison to Between Academic and Commercial Models

To answer RQ3, we examine the ordering and overlapping of the confidence intervals in the Figures 3.4-3.7. We note that we do not see signs of systemic differences between academic and commercial models in terms of their demographic disparities. When we examine the most biased model in each of the dataset and sociodemographic pairings, we observe no clear pattern.

Commercial models are most biased in skin type and lighting variables as well as on Adience and UTKFace in the perceived gender variable. Academic models are most biased on the CCD dataset in the perceived gender variable as well as every dataset except for CCD on the age variable. (They are tied in the other two instances). Thus, we conclude there is no systematic difference in the magnitude of the disparity exhibited by commercial and academic models writ large.

3.5 Implications and Hypotheses

Above, we have shown striking disparities in commercial facial analysis systems. These biases have potential for real harms felt by individuals. Facial *detection* is the first step in facial recognition. As such, the biases which we report here will propagate downstream into further facial analysis systems. Facial detection bias is the starting point for bias in other facial analyses, and research that addresses biases in detection will also serve any other facial analysis system which uses its outputs. However, downstream systems will still have their own biases.

Since we are external researchers, we can only speculate as to why these disparities and biases are observed since we do not have access to the models themselves. The biases for dark-skinned individuals and dimly-lit subjects is unfortunately aligned with many prior works on the subject. Among the reasons for this include luminance and pixel intensity, which unfortunately have been codified as being discriminatory against darker skinned people in photography for decades (Lewis, 2019).

On the other hand, the findings about older individuals and masculine-presenting individuals offer contrasting conclusions from existing work that audits facial analysis technologies. Regarding the finding the systems are more susceptible to noise-induced changes on masculine presenting subjects, we hypothesize that this might have to do with the size that a feminine-presenting subject's *head* takes up in an image. One gender presentation marker is hair and we hypothesize that the subject's entire head size might be a confounding factor in this bias phenomenon. We unfortunately do not have the data to test this hypothesis (since ground truth data for face detection includes just data on the face), but one could collect such data with sufficient ground truth.

3.6 Discussion & A Call to action

Revisiting our research questions, we come away with rather clear answers. We see that face detection models:

(**RQ1**): show that their robustness to noise could be improved significantly;

(**RQ2**): have significant perceived sociodemographic disparities in their performance on noise robustness tasks; and

(**RQ3**): show similar degrees of demographic bias across both academic and commercial models.

We believe that these results beget three main conclusions for different audiences who are interested in face detection systems and/or algorithmic bias. Our results suggest that commercial systems generally are no less biased on noise robustness than academic systems, for the types of noise corruptions we benchmarked. This is a rather striking result considering the resources large companies have at their disposal to tackle problems like demographic disparities in their products. Additionally, since demographic disparities in commercial products became a crucible following the publication of Buolamwini and Gebru (2018) in 2018, these corporations have had ample time to address and work towards solutions to these issues. While these companies have to varying degrees acknowledged the need to equal out demographic disparities in their products, we cannot fully know what investment they have placed on these issues, and specifically on disparities in noise robustness. So at this time, we can merely speculate.

If these companies have committed vast resources to address the demographic disparities in their products, and specifically in noise robustness, then our results lead us to conclude that these investments have generally not paid off. We conclude this because we now know that within each dataset and for most commercial model, there is at least one academic model which is at most as biased than it is. Further, since these academic models are published publicly with full source code and training procedures, we know that these models have not included any fairness constraints or considerations. Thus, if these companies *have* invested heavily in this problem, then we conclude that their investments have not paid off.

However, it is perhaps overly optimistic to think that corporations have invested in the mitigation of demographic bias in noise robustness — although we posit that this is not likely because many real-world use cases for facial analysis occur under imperfect "in-the-wild" conditions that would introduce various forms of natural noise. If in fact they have not done so, our results give a clear benchmark and goalpost for these corporations to improve. While in most cases, the commercial models are the most biased system, we should endeavor to expect that if these corporations plan to continue to publicly sell face detection software — a very socially and ethically provocative tool — that they should be investing in mitigating these biases and be able to do better than academic models which have no fairness considerations.

Our results add to the increasing body of research which finds various pernicious forms of demographic bias in facial recognition technologies. We provided strong evidence of the demographic biases present in face detection systems. We conclude that despite all the talk and publicity about concerns of demographic disparities in commercially provide products, large technology companies are no better at eliminating bias for noise robustness than academic models. Thus, we end this work with two broad calls to action:

To industry: This benchmark shows that the highly-resourced companies are no better than academic models at this robustness disparity in facial detection, a rather surprising comparison between where a trillion-dollar company could be—by spending a vanishing fraction of their liquid capital—and where it *should* be—where "should" is, admittedly, a value judgment, but a bipartisan one (Beyea and Kebede, 2021), and one gaining increasing traction in those firms' own home country (Ruane, 2021).

Our call to action, then, is as follows: pay attention to, work with, and fund academic research in unfairness in facial detection and noise, specifically natural and synthetic styles of noise. As our present work shows, academic models run hand-in-hand with—and, indeed, by some metrics beat—commercially deployed systems, and it would be of great benefit to further encourage unrestricted growth in that space, and to fertilize that growth with cross-boundary communication of techniques that have been tried internally at for-profit firms. Specific to our setting, both the present work and previous works (e.g., Buolamwini and Gebru, 2018; Raji and Buolamwini, 2019) would benefit immensely from at least partial access to the internal workings of commercial systems, including dataset curation processes. Beyond simply measuring disparities, the natural next step is to hypothesize reasons for those disparities and then to, at least partially, mitigate them via new techniques. Indeed, as this chapter shows, state-of-the-art academic models are arguably *beating* commercial models in some ways, so the value within this communication would flow both ways. Without a clear line of communication between academic and industrial researchers, this latter process is hampered.

To the public sector: The public sector provides a great service in both impacting the evolution of, and creating as well as enforcing the present state of social and legal norms. For example, in the United States, for our specific setting, the National Institute of Standards and Technology (NIST) Face Recognition Vendor Test (FRVT) has measured and monitored progress in both commercial and academic facial analysis systems. It has been run for at least the last two decades, and has been updated numerous times. Indeed, in a recent FRVT Update, NISTIR 8280 (2019), NIST brought demographic concerns into the forefront. NIST's venerable FRVT has a history of incorporating natural noise into its barrage of tests; we would ask NIST, and analogous non-regulatory and standards-settings bodies in other countries, to consider updating their tests (e.g., FRVT) to include the cross section of bias and forms of noise. Our work motivates the need for monitoring in this area.

To the regulatory side, we are encouraged by and seek further acceptance of results publicized by both academics and industrial researchers. Washington State aims to set an example here with its recently enacted State Bill 6281, which states "if the results of . . . independent testing identify material unfair performance differences across subpopulations . . . then the processor must develop and implement a plan to address the identified performance differences" (of Washington 66th Legislature 2020 Regular Session, 2020). We believe that this benchmark meets this definition and hope the public sector has a robust enforcement mechanism for such legislation. We encourage other researchers to continue to audit existing commercial products, and believe our approach to compare commercial models to academic models enriches the scholarly and social discourse about facial recognition technology.

> Out beyond ideas of wrongdoing and
> rightdoing,
> there is a field. I'll meet you there.
> When the soul lies down in that grass,
> the world is too full to talk about.
> Ideas, language, even the phrase *each*
> *other*
> doesn't make any sense.
>
> *A Great Wagon*
>
> Rumi

Chapter 4: Rethinking Bias Mitigation: Fairer Architectures Make for Fairer Face Recognition

This work was done in collaboration with my co-first author, Rhea Sukthanker, and Colin White, John P. Dickerson, Frank Hutter, and Micah Goldblum.

Conventional belief in the fairness community is that one should first find the highest performing model for a given problem and then apply a bias mitigation strategy. One starts with an existing model architecture and hyperparameters, and then adjusts model weights, learning procedures, or input data to make the model fairer using a pre-, post-, or in-processing bias mitigation technique. Motivated by the belief that the inductive bias of a model architecture is more important than the bias mitigation strategy, we take a different approach to bias mitigation. We show that finding an *a*rchitecture that is more fair offers significant gains compared to conventional bias mitigation strategies in the domain of face recognition, a task that is notoriously challenging to de-bias. To this end, we conduct the first neural architecture search for fairness, jointly with a search for hyperparameters. Our search outputs a suite of models which Pareto-dominate all other competitive architectures in terms of accuracy and fairness on the two most widely used datasets for face identification: CelebA and VGGFace2. This work challenges the assumption that bias mitigation pipelines should default to existing popular architectures which were optimized for accuracy — instead we show that it may be more beneficial to begin with a fairer architecture as the foundation of such pipelines.

4.1 Introduction

Machine learning is applied to a wide variety of socially-consequential domains, e.g., credit scoring, fraud detection, hiring decisions, criminal recidivism, loan repayment, and face recognition (Mukerjee et al., 2002; Ngai et al., 2011; Learned-Miller et al., 2020; Barocas et al., 2017), with many of these applications impacting the lives of people more than ever — often in biased ways (Buolamwini and Gebru, 2018; Joo and Kärkkäinen, 2020; Wang et al., 2020b). Dozens of formal definitions of fairness have been proposed (Narayanan, 2018), and many algorithmic techniques have been developed for debiasing according to these definitions (Verma and Rubin, 2018). Many debiasing algorithms fit into one of three (or arguably four (Savani et al., 2020)) categories: pre-processing (e.g., Feldman et al., 2015b; Ryu et al., 2018; Quadrianto et al., 2019; Wang and Deng, 2020), in-processing (e.g., Zafar et al., 2017b, 2019; Donini et al., 2018;

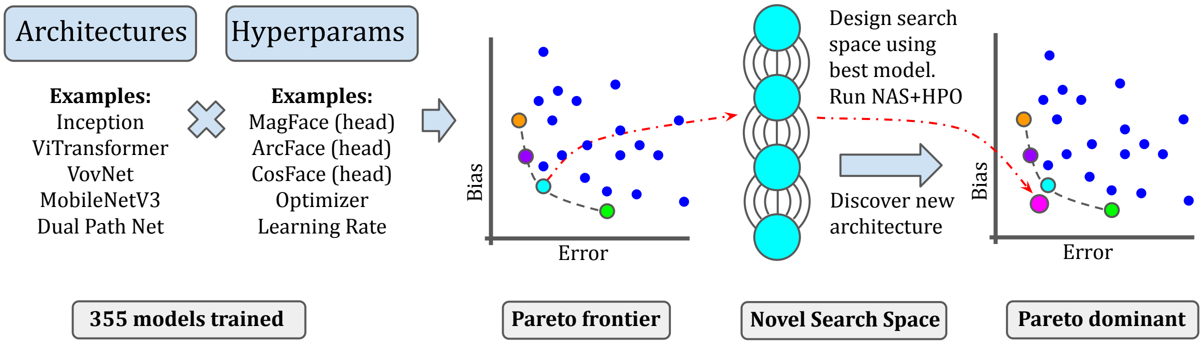


Figure 4.1: Overview of our methodology.

Goel et al., 2018; Padala and Gujar, 2020; Wang and Deng, 2020; Martinez et al., 2020; Nanda et al., 2021; Diana et al., 2020; Lahoti et al., 2020), or post-processing (e.g., Hardt et al., 2016; Wang et al., 2020b).

Conventional wisdom is that in order to effectively mitigate bias, we should start by selecting a model architecture and set of hyperparameters which are optimal in terms of accuracy and then apply a mitigation strategy to reduce bias while minimally impacting accuracy. While existing methods for de-biasing machine learning systems use a fixed neural architecture and hyperparameter setting, we instead ask a fundamental question which has received little attention: *how much does model-bias arise from the architecture and hyperparameters?* We further ask whether we can we exploit the extensive research in the fields of neural architecture search (NAS) (Elsken et al., 2019) and hyperparameter optimization (HPO) (Feurer and Hutter, 2019) to search for more inherently fair models.

We demonstrate our results on face identification systems where pre-, post-, and in-processing techniques have fallen short of de-biasing face recognition systems, and training fair models in this setting demands addressing several technical challenges (Cherepanova et al., 2023). Face identification is a type of face recognition which is regularly deployed across the world by government agencies for tasks including surveillance, employment, and housing decisions. Face recognition systems exhibit disparity in accuracy based on race and gender (Grother et al., 2019; Raji et al., 2020; Raji and Fried, 2021; Learned-Miller et al., 2020). For example, some face recognition models are 10 to 100 times more likely to give false positives for Black or Asian people, compared to white people (Allyn, 2020). This bias has already led to multiple false arrests and jail time for innocent Black men in the USA (Hill, 2020a).

In this work, we conduct the first large-scale analysis of the relationship between hyperparameters, architectures, and bias. We train a diverse set of 29 architectures, ranging from ResNets (He et al., 2016b) to vision transformers (Dosovitskiy et al., 2021; Liu et al., 2021) to Gluon Inception V3 (Szegedy et al., 2016) to MobileNetV3 (Howard et al., 2019). We conduct these experiments, for a total of 88 493 GPU hours, on the two most widely used datasets in face identification that have sociodemographic labels: CelebA (Liu et al., 2015) and VGGFace2 (Cao et al., 2018). We train each of these architectures across different combinations of head, optimizer, and learning rate.

Next, we exploit this observation in order to design architectures with a better fairness-accuracy tradeoff. We initiate the study of NAS for fairness by conducting the first use of NAS+HPO to jointly optimize fairness and accuracy. To tackle this problem, we construct a search space based on the highest-performing architecture from our analysis. We adapt the existing Sequential Model-based Algorithm Configuration method (SMAC) (Lindauer et al., 2022) for multi-objective architecture and hyperparameter search. We discover a Pareto frontier of face recognition models that outperform existing state-of-the-art models on both accuracy and multiple fairness metrics. An overview of our methodology can be found in 4.1. We release all of our code and raw results here, so that users can adapt our work to any bias measure of their choice.

We summarize our primary contributions:

- We provide a new class of bias mitigation strategies. We identify that architectures have a profound influence on fairness, and then we exploit that insight in order to design fairer architectures via Neural Architecture Search and Hyperparameter Optimization.
- Our observations show that it is better to search for a fairer architecture than it is to adjust an unfair one. We conclude that the implicit convention of choosing the highest-accuracy architectures is a sub-optimal strategy, and we suggest that architectures and hyperparameters play a significant role in determining the best fairness-accuracy tradeoff.
- Our approach finds architectures which are Pareto-optimal on a variety of fairness metrics on both CelebA and VGGFace2. Additionally, when comparing to other bias mitigation techniques, our approach remains Pareto-optimal, finding the fairest model.

4.2 Background and Related Work

4.2.1 Face Identification.

Face recognition tasks fall into two categories: verification and identification. We focus on face *identification* which asks whether a given person in a source image appears within a gallery composed of many target identities and their associated images; this is a one-to-many comparison. Novel techniques in face recognition tasks, such as ArcFace (Wang et al., 2018), CosFace (Deng et al., 2019), and MagFace (Meng et al., 2021), use deep networks (often called the *backbone*) to extract feature representations of faces and then compare those to match individuals (with mechanisms called the *head*). Generally, *backbones* take the form of image feature extractors and *heads* resemble MLPs with specialized loss functions. Often, the term "head" refers to both the last layer of the network and the loss function. We focus our analysis on identification, and we focus our evaluation on examining how close images of similar identities are in the feature space of trained models, since the technology relies on this feature representation to differentiate individuals. An overview of recent research on these topics can be found in Wang and Deng (2018).

4.2.2 Bias Mitigation in Face Recognition.

The existence of differential performance of face recognition on population groups and subgroups has been explored in a variety of settings. Earlier work (e.g., Klare et al., 2012; O'Toole et al., 2012) focuses on single-demographic effects (specifically, race and gender) in pre-deep-learning face detection and recognition. Buolamwini and Gebru (2018) uncover unequal performance at the phenotypic subgroup level in, specifically, a gender classification task powered by commercial systems. Raji and Buolamwini (2019) provide a follow-up analysis – exploring the impact of the public disclosures of Buolamwini and Gebru (2018) – where they find that named companies (IBM, Microsoft, and Megvii) updated their APIs within a year to address some concerns that had surfaced. Further research continues to show that commercial face recognition systems still have sociodemographic disparities in many complex and pernicious ways (Drozdowski et al., 2020; Dooley et al., 2021b; Jaiswal et al., 2022; Dooley et al., 2022c; Jaiswal et al., 2022).

Facial recognition is a large and complex space with many different individual technologies, some with bias mitigation strategies designed just for them (Leslie, 2020; Wu et al., 2020). The main bias mitigation strategies for facial identification are described in 4.5.

4.2.3 Neural Architecture Search (NAS) and Hyperparameter Optimization (HPO).

Deep learning derives its success from the manually designed feature extractors which automate the feature engineering process. Neural architecture search (NAS) (Elsken et al., 2019), on the other hand, aims at automating the very design of network architectures for a task at hand. NAS can be seen as a subset of HPO (Feurer and Hutter, 2019), which refers to the automated search for optimal hyperparameters, such as learning rate, batch size, dropout, loss function, optimizer, and architectural choices. Rapid and extensive research on NAS for image classification and object detection has been witnessed as of late (Liu et al., 2018a; Zela et al., 2019; Xu et al., 2019; Pham et al., 2018; Cai et al., 2018). Deploying NAS techniques in face recognition systems has also seen a growing interest (Zhu, 2019; Wang, 2021). For example, reinforcement learning-based NAS strategies (Xu et al., 2019) and one-shot NAS methods (Wang, 2021) have been deployed to search for an efficient architecture for face recognition with low *error*. However, in a majority of these methods, the training hyperparameters for the architectures are *fixed*, which we observe should be reconsidered in order to obtain the fairest possible face recognition systems. Moreover one-shot NAS methods have also been applied for multi-objective optimization (Guo et al., 2020; Cai et al., 2019), e.g., optimizing accuracy and size. However, none of these methods can be applied for a joint architecture and hyperparameter search.

For the case of tabular datasets, a few works have applied hyperparameter optimization to mitigate bias in models. Perrone et al. (2021) introduces a Bayesian optimization framework to optimize accuracy of models while satisfying a bias constraint. The concurrent works of Schmucker et al. (2020) and Cruz et al. (2020) extend Hyperband (Li et al., 2017) to the multi-objective setting and show its applications to fairness. The former was later extended to the asynchronous setting (Schmucker et al., 2021). Lin et al. (2022) proposes de-biasing face recognition models through model pruning. However, they consider just two architectures and just one set of hyperparameters. To the best of our knowledge, no prior work uses any AutoML technique (NAS, HPO, or joint NAS and HPO) to design fair face recognition models, and no prior work uses NAS to design fair models for any application.

4.3 Architectures and Hyperparameters: A Case Study

In this section, we give an overview of our Neural Architecture Search-based bias mitigation technique as well as the experiments we ran in order to answer: *are architectures and hyperparameters important for fairness?* To this end, we conduct an exploration of many different model architectures using different hyperparameter combinations. We find strong evidence that accuracy is not predictive of fairness metrics (4.7 in the appendix), which motivates us to use NAS techniques to optimize fairness and accuracy *jointly*. We explore this in 4.4.

4.3.1 NAS-based Bias Mitigation

We propose to find fairer models and architectures not by using pre-, post-, or in-processing techniques but by searching for and identifying a set of architectures and hyperparameters which are fairer than the baseline. The first step of this methodology starts with identifying a search space of architectures and hyperparameters of candidate models and then deploying NAS+HPO. In the domain of face identification, we design a search space around the Dual Path Network (DPN) (Chen et al., 2017) architecture, and detail why we made that decision in 4.3.2. In general, a domain expert using our NAS-based bias mitigation technique would be able to establish their search space immediately. We discuss the details of the second step in 4.4.

4.3.2 Architectures and Hyperparameter Experiments

4.3.3 Experimental Setup.

We train and evaluate each model configuration on a gender-balanced subset of the two most popular face identification datasets: CelebA and VGGFace2. CelebA (Liu et al., 2015) is a large-scale face attributes dataset with more than 200K celebrity images and a total of 10 177 gender-labeled identities. VGGFace2 (Cao et al., 2018) is a much larger dataset designed specifically for face identification and comprises over 3.1 million images and a total of 9 131 gender-labeled identities. While this work analyzes phenotypic metadata (perceived gender), the reader should not interpret our findings absent a social lens of what these demographic groups mean inside society. We guide the reader to Hamidi et al. (2018) and Keyes (2018) for a look at these concepts for gender.

To study the importance of architectures and hyperparamters for fairness, we use the following training pipeline – ultimately conducting 355 training runs with different combinations of 29 architectures from the Pytorch Image Model (`timm`) database (Wightman, 2019) and hyperparameters.

The list of the models we study from `timm` are: `coat_lite_small` (Xu et al., 2021), `convit_base` (d'Ascoli et al., 2021), `cspdarknet53` (Wang et al., 2020a), `dla102x2` (Yu et al., 2018), `dpn107` (Chen et al., 2017), `ese_vovnet39b` (Lee and Park, 2020), `fbnetv3_g` (Dai et al., 2021), `ghostnet_100` (Han et al., 2020b), `gluon_inception_v3` (Szegedy et al., 2016), `gluon_xception65` (Chollet, 2017), `hrnet_w64` (Sun et al., 2019), `ig_resnext101_32x8d` (Xie et al., 2016), `inception_resnet_v2` (Szegedy et al., 2017), `inception_v4` (Szegedy et al., 2017), `jx_nest_base` (Zhang et al., 2021), `legacy_senet154` (Hu et al., 2018), `mobilenetv3_large_100` (Howard et al., 2019), `resnetrs101` (Bello et al., 2021), `rexnet_200` (Han et al., 2020a), `selecsls60b` (Mehta et al., 2019), `swin_base_patch4_window7_224` (Liu et al., 2021), `tf_efficientnet_b7_ns` (Tan and Le, 2019), `tnt_s_patch16_224` (Han et al., 2021), `twins_svt_large` (Chu et al., 2021), `vgg19` (Simonyan and Zisserman, 2015), `vgg19_bn` (Simonyan and Zisserman, 2015), `visformer_small` (Chen et al., 2021), `xception` and `xception65` (Chollet, 2017).

We conduct training runs with both the default hyperparameters as well as hyperparameters which are standardized across all architecutres, e.g., AdamW (Loshchilov and Hutter, 2019) with lr=0.001 and SGD with lr=0.1. For each model, we use the default learning rate and optimizer that was published with that model. We then conduct a training run with these hyperparameters for each of three heads, ArcFace (Wang et al., 2018), CosFace (Deng et al., 2019), and MagFace (Meng et al., 2021). Next, we use that default learning rate with both AdamW and SGD optimizers (again with each head). Finally, we also conduct training routines with AdamW and SGD with unified learning rates (SGD with lr=0.1 and AdamW with lr=0.001). In total, we run a single architecture between 9 and 13 times (9 times if the default optimizer and learning rates were the same as the standardized, and 13 times otherwise). All other hyperparameters were the same for each model training run.

We study at most 13 configurations per model ie 1 default configuration corresponding to the original model hyperparameters with CosFace as head. Further, we have at most 12 configs consisting of the 3 heads (CosFace, ArcFace, MagFace) × 2 learning rates(0.1,0.001) × 2 optimizers (SGD, AdamW). All the other hyperparameters are held constant for training all the models. All model configurations are trained with a total batch size of 64 on 8 RTX2080 GPUS for 100 epochs each.

4.3.4 Evaluation procedure.

The standard approach in face identification tasks is to evaluate the performance of the learned representations. Recall that face recognition models usually learn representations with an image backbone and then learn a mapping from those representations onto identities of individuals with the head of the model. As is commonplace (Cherepanova et al., 2021, 2022), evaluating the learned feature representations allows us to better isolate the impact of the image backbone architecture. We break each dataset into train, validation, and test sets. We conduct our search for novel architectures using the train and validation splits, and then show the improvement of our model on the test set.

The main performance metric for the models will be representation error, which we will henceforth simply refer to as *Error*. Recall that we pass each test image through a trained model and save the learned representation. To compute *Error*, we merely ask, for a given probe image/identity, whether the closest image in feature space is *not* of the same person based on l_2 distance.

The most widely used fairness metric in face identification is *rank disparity*, which is explored in the NIST FRVT (Patrick J. Grother, 2010). To compute the rank of a given image/identity, we ask how many images of a different identity are closer to the image in feature space. We define this index as the rank of a given image under consideration. Thus, Rank(image) $= 0$ if and only

Table 4.1: The fairness metrics explored in this chapter. Rank Disparity is explored in the main paper and the other metrics are reported in 4.5.2

Fairness Metric	Equation
Rank Disparity	$\lvert\text{Rank}(male) - \text{Rank}(female)\rvert$
Disparity	$\lvert\text{Accuracy}(male) - \text{Accuracy}(female)\rvert$
Ratio	$\lvert 1 - \frac{\text{Accuracy}(male)}{\text{Accuracy}(female)}\rvert$
Rank Ratio	$\lvert 1 - \frac{\text{Rank}(male)}{\text{Rank}(female)}\rvert$
Error Ratio	$\lvert 1 - \frac{\text{Error}(male)}{\text{Error}(female)}\rvert$

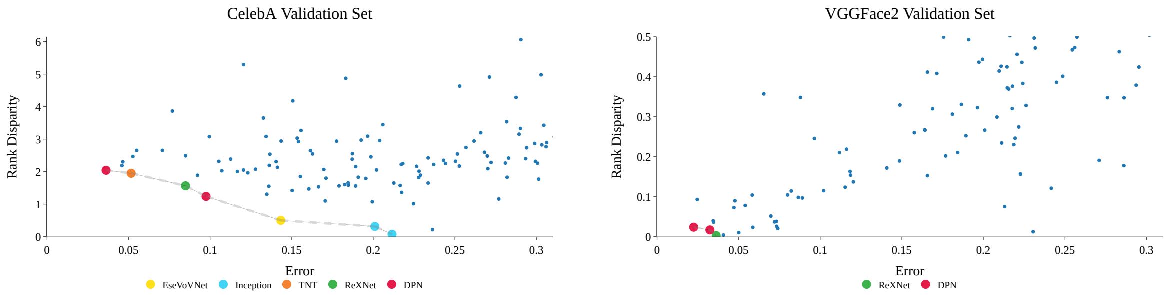


Figure 4.2: (Left) CelebA (Right) VGGFace2. Error-Rank Disparity Pareto front of the architectures with lowest error (< 0.3). Models in the lower left corner are better. The Pareto front is notated with a dashed line. Other points are architecture and hyperparameter combinations which are not Pareto-optimal.

if Error(image) $= 0$; Rank(image) > 0 if and only if Error(image) $= 1$. We examine the **rank disparity**: the absolute difference of the average ranks for each perceived gender in a dataset $\mathcal{D}$:

$$\left| \frac{1}{|\mathcal{D}_{\text{male}}|} \sum_{x \in \mathcal{D}_{\text{male}}} \text{Rank}\,(x) - \frac{1}{|\mathcal{D}_{\text{female}}|} \sum_{x \in \mathcal{D}_{\text{female}}} \text{Rank}(x) \right|. \tag{4.1}$$

We focus on rank disparity throughout the main body of this chapter as it is the most widely used in face identification, but we explore other forms of fairness metrics in face recognition as well. We study these models across five important fairness metrics in face identification: Rank Disparity, Disparity, Ratio, Rank Ratio, and Error Ratio. Each of these metrics is defined in Table 4.1.

4.3.5 Results and Discussion.

By plotting the performance of each training run on the validation set with the error on the x-axis and rank disparity on the y-axis in 4.2, we can easily conclude two main points. First, optimizing for error does not always optimize for fairness, and second, different architectures have different fairness properties. We also find the DPN architecture has the lowest error and is Pareto-optimal on both datasets; thus, we will use that architecture to design our search space in 4.4.

On the first point, a search for architectures and hyperparameters which have high performance on traditional metrics does not translate to high performance on fairness metrics. We see that within models with lowest error – those models which are most interesting to the community – there is low correlation between error and rank disparity (for models with error < 0.3, $\rho = .113$ for CelebA and $\rho = .291$ for VGGFace2). We do note however, the differences between the two datasets at the most extreme low errors. First, for VGGFace2, the baseline models already have very low error, with there being 10 models with error less than 0.05; CelebA only has three such models. Additionally, models with low error also have low Rank Disparity on VGGFace2 whereas this is not the case on CelebA; however, we foreshadow that this is more true on the validation set than it is on the test set. This can be seen by looking at the Pareto curves in 4.2.

We see that optimizing architectures and hyperparameters for error alone will not lead to fair models. The Pareto optimal models on CelebA are versions of DPN, TNT, ReXNet, VovNet, and ResNets. The Pareto optimal models on VGGFace2 are DPN and ReXNet. Further, different architectures exhibit different optimal hyperparameters. For example on CelebA, the Xception65 architecture finds SGD with ArcFace and AdamW with ArcFace are Pareto-optimal whereas the

Inception-ResNet architecture finds MagFace and CosFace optimal with SGD.

4.4 Neural Architecture Search for Bias Mitigation

In this section, we employ joint NAS+HPO as a bias mitigation strategy. Inspired by our findings on the importance of architecture and hyperparameters for fairness in 4.3, we initiate the first joint NAS+HPO study for fairness in face recognition. We start by describing our search space and search strategy. We then present a comparison between the architectures obtained from our NAS-based mitigation strategy and other popular face recognition bias mitigation strategies. We conclude that our joint NAS+HPO indeed discovers simultaneously accurate and fair architectures.

4.4.1 Search Space Design and Search Strategy

We design our search space based on our analysis in 4.3, specifically around the Dual Path Networks (Chen et al., 2017) architecture which has the lowest error and is Pareto-optimal on both datasets. In particular, our search space is inspired by DPN due to its strong trade-off between rank disparity and accuracy as seen in 4.2.

4.4.2 Hyperparameter Search Space Design.

We choose three categories of hyperparameters for NAS+HPO: the architecture head/loss, the optimizer, and the learning rate (4.2).

4.4.3 Architecture Search Space Design.

Dual Path Networks (Chen et al., 2017) for image classification share common features (ResNets (He et al., 2016a)) while possessing the flexibility to explore new features (Huang et al., 2017) through a dual path architecture. We replace the repeating `1x1_conv-3x3_conv-1x1_conv` block with a simple recurring searchable block. depicted in 4.6. Furthermore, we stack multiple such searched blocks to closely follow the architecture of Dual Path Networks. We have nine possible choices for each of the three operations in the DPN block, each of which we give a number 1 through 9, depicted in 4.6. The choices include a vanilla convolution, a convolution with pre-normalization and a convolution with post-normalization

4.4.4 Obtained architectures and hyperparameter configurations from Black-Box-Optimization

In 4.3 we present the architectures and hyperparameters discovered by SMAC. Particularly we observe that `conv 3x3` followed `batch norm` is a preferred operation and CosFace is the preferred head/loss choice.

We thus have 81 different architectures which can be searched (in addition to an infinite number of hyperparameter sets). We denote each of these architectures by `XYZ` where $X, Y, Z \in [9]$, i.e., the architecture `180` would represent the architecture which has the first operation, followed by the eighth, followed by the zeroth.

To summarize, our search space consists of a choice among 81 different architecture types, three different head types, three different optimizers, and a infinite number of choices for the

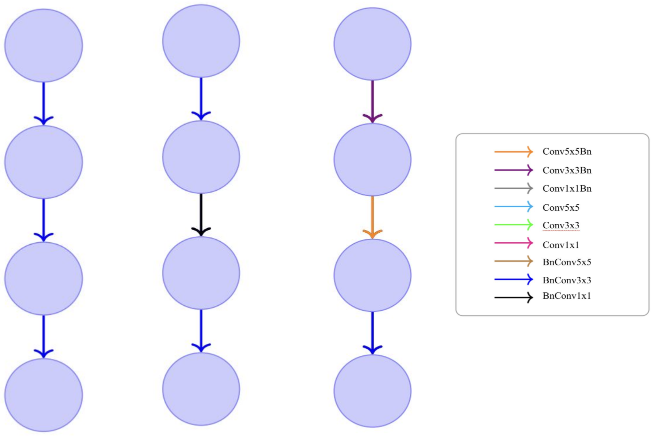


Figure 4.3: SMAC discovers the above building blocks with (a) corresponding to architecture with CosFace, with SGD optimizer and learning rate of 0.2813 as hyperparamters (b) corresponding to CosFace, with SGD as optimizer and learning rate of 0.32348 and (c) corresponding to CosFace, with AdamW as optimizer and learning rate of 0.0006

learning rate.

We navigate the search space using Black-Box-Optimization (BBO) with the following desiderata:

4.4.5 Multi-fidelity optimization.

A single function evaluation for our use-case corresponds to training a deep neural network on a given dataset. This is generally quite expensive for traditional deep neural networks on moderately large datasets. Hence we would like to use cheaper approximations to speed up the black-box algorithm with multi-fidelity optimization techniques (Schmucker et al., 2021; Li et al., 2017; Falkner et al., 2018), e.g., by evaluating many configurations based on short runs with few epochs and only investing more resources into the better-performing ones.

4.4.6 Multi-objective optimization.

We want to observe a trade-off between the accuracy of the face recognition system and the fairness objective of choice, so our joint NAS+HPO algorithm supports multi-objective optimization (Paria et al., 2020; Davins-Valldaura et al., 2017; Mao-Guo et al., 2009).

The SMAC3 package (Lindauer et al., 2022) offers a robust and flexible framework for Bayesian Optimization with few evaluations. SMAC3 offers a SMAC4MF facade for *multi-fidelity optimization* to use cheaper approximations for expensive deep learning tasks like ours. We choose ASHA (Schmucker et al., 2021) for cheaper approximations with the initial and maximum fidelities set to 25 and 100 epochs, respectively, and $\eta = 2$. Every architecture-hyperparameter configuration evaluation is trained using the same training pipeline as in 4.3. For the sake of simplicity, we use the ParEGO (Davins-Valldaura et al., 2017) algorithm for *multi-objective optimization* with ρ set to 0.05.

4.5 Results

Recall our main motivation is to demonstrate that our NAS-based bias mitigation strategy is comparative to or better than other strategies in reducing bias in facial identification. We proceed in two steps: first, we discuss the model found with our NAS+HPO approach, and then we compare it to other mitigation baselines:

4.5.1 Novel architectures

We conducted one NAS+HPO search for each dataset by searching on the train and validation sets. After running these searches, we identified three new candidate architectures for CelebA

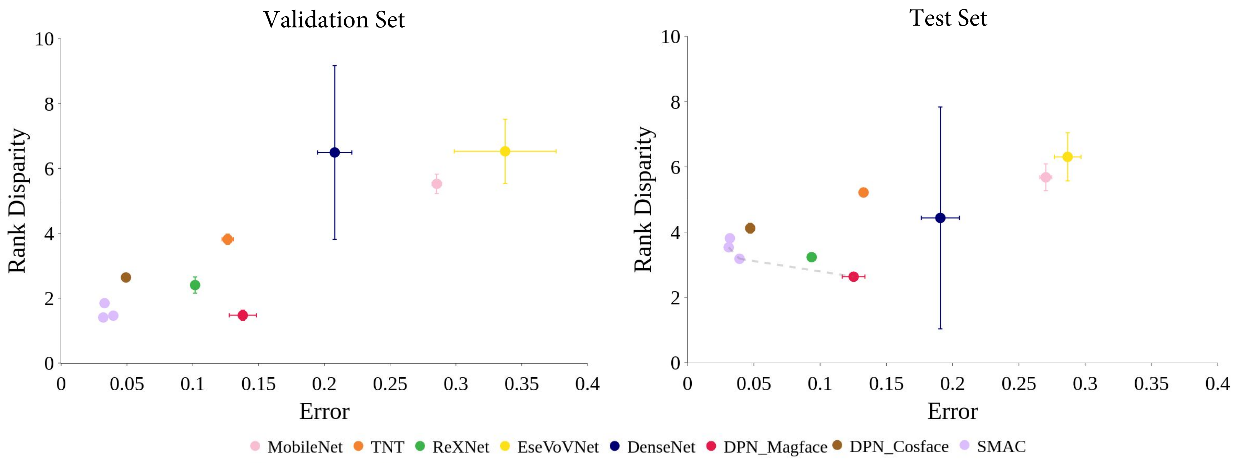


Figure 4.4: Pareto front of the models discovered by SMAC and the rank-1 models from `timm` for the *(a)* validation and *(b)* test sets on CelebA. Each point corresponds to the mean and standard error of an architecture after training for 3 seeds. The SMAC models Pareto-dominate the top performing `timm` models ($Error < 0.1$).

(SMAC_000, SMAC_010, and SMAC_680), and one candidate for VGGFace2 (SMAC_301) where the naming convention follows that described in 4.4.1. We then retrained each of these models and those high performing models from 4.3 for three seeds to study the robustness of error and disparity for the models; we evaluated their performance on the validation and test sets for each dataset, where we follow the evaluation scheme of 4.3.

On CelebA (4.4), our models Pareto-dominate all models with nontrivial accuracy on the val set. On the test set, our models still Pareto-dominate all highly competitive models (with Error<0.1), but one of the original configurations (DPN with Magface) also becomes Pareto-optimal. However, the error of this architecture is 0.13, which is significantly higher than our models (0.03-0.04).

On VGGFace2 (4.5), our models are Pareto-optimal for both the validation and test sets. On the test set of VGGFace2 it is much harder to achieve low disparity, yet our model improves upon the other baselines. Furthermore, from 4.4 it is also apparent that some models such as VoVNet

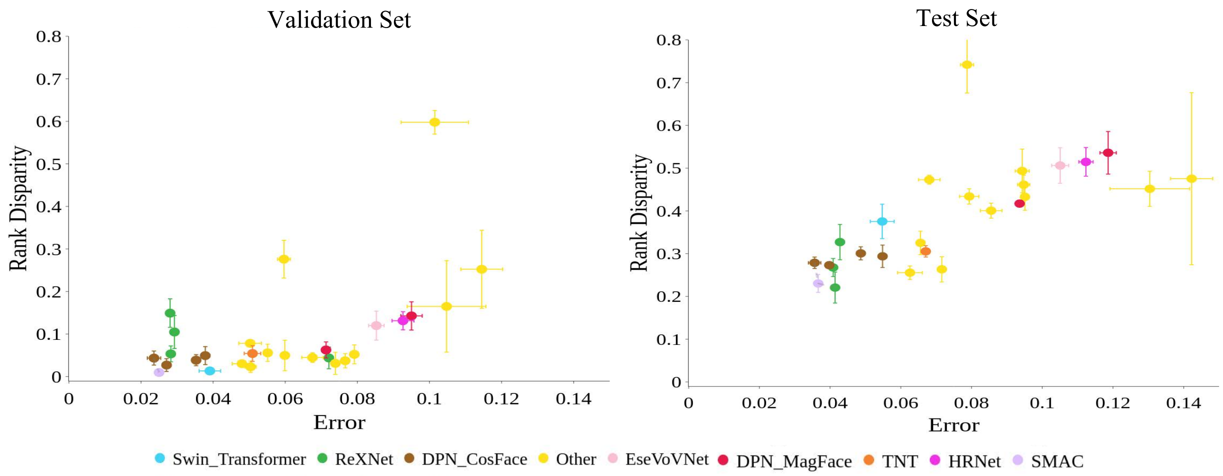


Figure 4.5: Pareto front of the models discovered by SMAC and the rank-1 models from `timm` for the *(a)* validation and *(b)* test sets on VGGFace2. Each point corresponds to the mean and standard error of an architecture after training for 3 seeds. The SMAC models Pareto-dominate the top performing `timm` models (Error<0.1).

and DenseNet show very large standard errors across seeds. Hence, it becomes very important to also study the robustness of the models across seeds along with the accuracy and disparity Pareto front.

We also compare to the current state of the art baseline ArcFace (Deng et al., 2019), which, using our training pipeline on CelebA data with face identification as our task, achieves an error of 4.35%. Our best performing novel architecture, however, achieves an error of 3.10%, clearly outperforming ArcFace. Similarly, the current VGGFace2 state of the art baseline (Wang et al., 2021) achieves an error of 4.5% and our best performing novel architecture achieves a much lower error of 3.66%.

4.5.2 Analysis of the Pareto-Front of different Fairness Metrics

In this section, we include additional plots that support and expand on the main paper. Primarily, we provide further context of the Figures in the main body in two ways. First, we

Table 4.2: Searchable hyperparameter choices.

Hyperparameter	Choices
Architecture Head/Loss	MagFace, ArcFace, CosFace
Optimizer Type	Adam, AdamW, SGD
Learning rate (conditional)	Adam/AdamW → $[1e-4, 1e-2]$, SGD → $[0.09, 0.8]$

Table 4.3: Operation choices and definitions.

Operation Index	Operation	Definition
0	BnConv1x1	Batch Normalization → Convolution with 1x1 kernel
1	Conv1x1Bn	Convolution with 1x1 kernel → Batch Normalization
2	Conv1x1	Convolution with 1x1 kernel
3	BnConv3x3	Batch Normalization → Convolution with 3x3 kernel
4	Conv3x3Bn	Convolution with 3x3 kernel → Batch Normalization
5	Conv3x3	Convolution with 3x3 kernel
6	BnConv5x5	Batch Normalization → Convolution with 5x5 kernel
7	Conv5x5Bn	Convolution with 5x5 kernel → Batch Normalization
8	Conv5x5	Convolution with 5x5 kernel

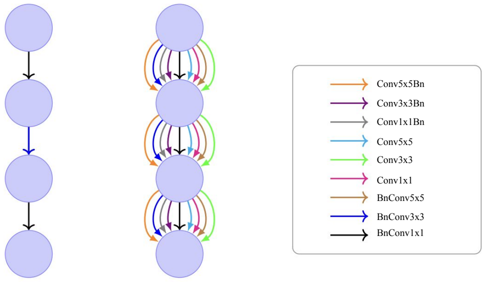


Figure 4.6: DPN block (left) vs. our searchable block (right).

provide replication plots of the figures in the main body but for all models. Recall, the plots in the main body only show models with Error<0.3, since high performing models are the most of interest to the community. Second, we also show figures which depict other fairness metrics used in facial identification. The formulas for these additional fairness metrics can be found in 4.1.

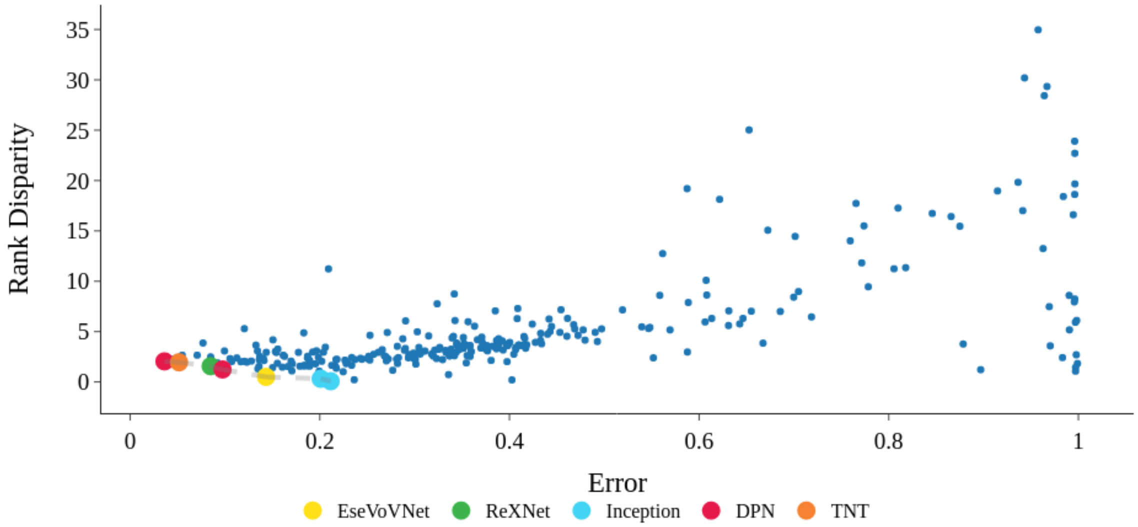


Figure 4.7: Replication of CelebA 4.2 with all data points. Error-Rank Disparity Pareto front of the architectures with any non-trivial error. Models in the lower left corner are better. The Pareto front is notated with a dashed line. Other points are architecture and hyperparameter combinations which are not Pareto-dominant.

We replicate 4.2 in 4.7 and 4.8. We add additional metrics for CelebA in 4.9-4.11 and for VGGFace in 4.12-4.15.

4.5.3 Novel Architectures Outperform other Bias Mitigation Strategies

There are three common pre-, post-, and in-processing bias mitigation strategies in face identification. First, Chang et al. (2020) demonstrated that randomly flipping labels in the training data of the subgroup with superior accuracy can yield more fair systems; we call this technique `Flipped`. Next, Wang and Deng (2020) use different angular margins during training and therefore promoting better feature discrimination for the minority class; we call this technique `Angular`. Finally, Morales et al. (2020) introduced `SensitiveNets` which is a sensitive information removal network trained on top of a pre-trained feature extractor with an adversarial

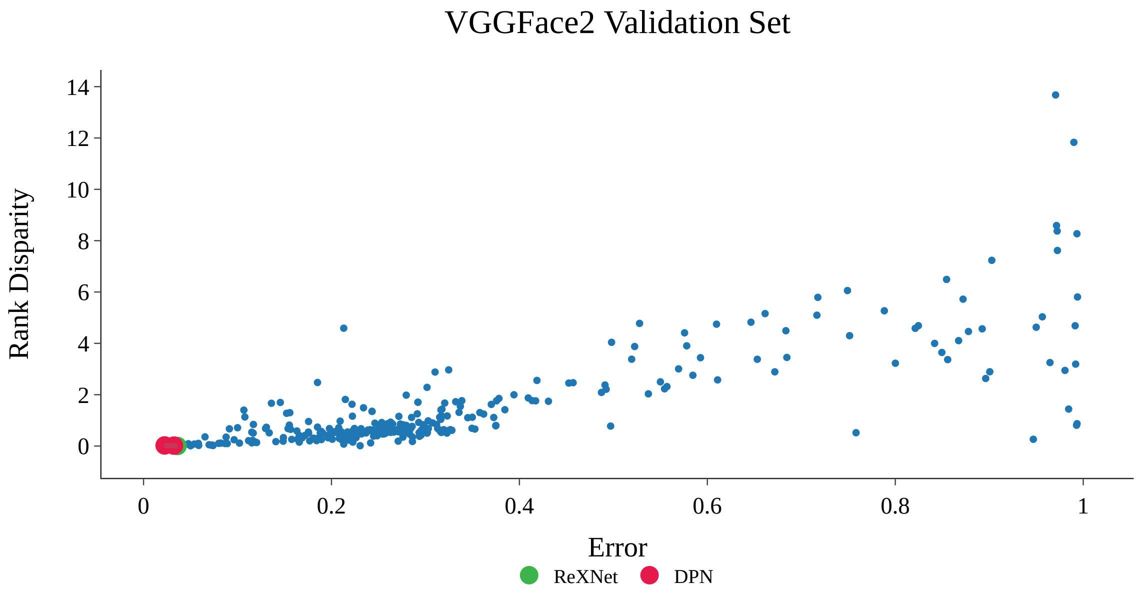


Figure 4.8: Replication of VGGFace2 4.2 with all data points. Error-Rank Disparity Pareto front of the architectures with any non-trivial error. Models in the lower left corner are better. The Pareto front is notated with a dashed line. Other points are architecture and hyperparameter combinations which are not Pareto-dominant.

sensitive regularizer. While other bias mitigation techniques exist in face recognition, these are the most used and pertient to *face identification*. See Cherepanova et al. (2023) for an overview of the technical challenges of bias mitigation in face recognition. We take top performing, Pareto-optimal models from the previous section and apply the three bias mitigation techniques: `Flipped`, `Angular`, and `SensitiveNets`. We also apply these same techniques to the novel architectures that we found. We report results in 4.6.

Critically, we observe that the novel architectures we found with our NAS-based approach Pareto-dominate the bias-mitigated models. In VGGFace2, the SMAC_301 model achieves the best performance, both in terms of error and fairness, when comparing to the bias-mitigated models. On CelebA, the same is true for the SMAC_680 model.

Additionally, we combined the three other bias mitigation methods with our SMAC models, in other words, we conducted our NAS approach and then applied the `Flipped`, `Angular`,

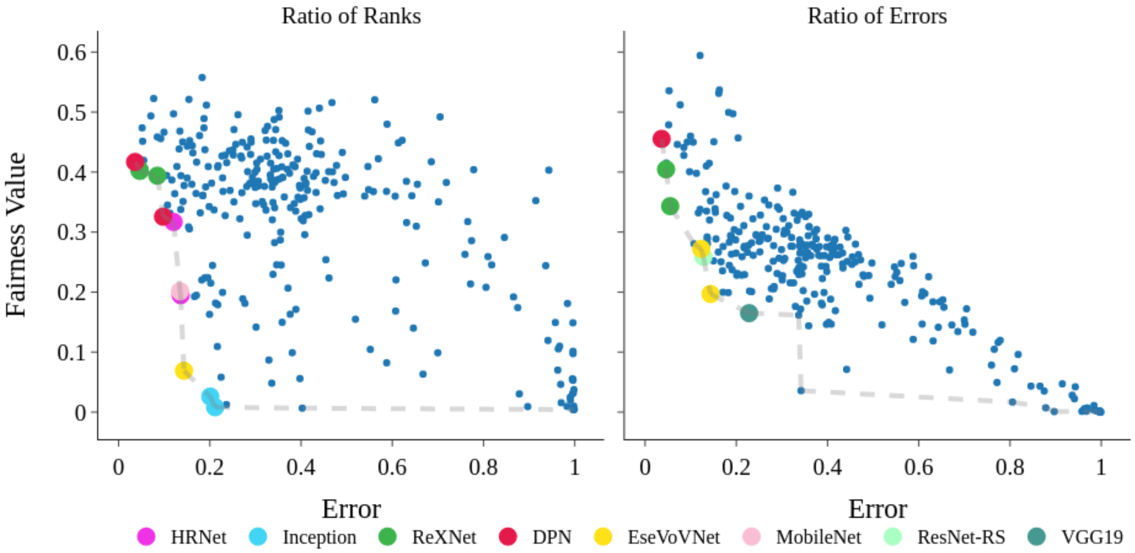


Figure 4.9: Replication of 4.7 on the CelebA validation dataset with Ratio of Ranks (left) and Ratio of Errors (right) metrics.

and `SensitiveNets` approach as well. We see that on both datasets, we continue to Pareto-dominate the other bias mitigation strategies. These techniques ultimately yield the model with the lowest rank disparity of all the models (0.18 on VGGFace2 and 0.03 on CelebA). In particular, the bias improvement of SMAC_000+`Flipped` model is notable, achieving a score of 0.03 whereas the lowest rank disparity of any model from 4.4 is 2.63, a 98.9% improvement.

In the next subsection, we demonstrate that this result is robust to the fairness metric — specifically our bias mitigation strategy Pareto-dominates the other approaches on all five fairness metrics.

4.5.4 Comparison to other Bias Mitigation Techniques on all Fairness Metrics

We have shown that our bias mitigation approach Pareto-dominates the existing bias mitigation techniques in face identification on the Rank Disparity metric. Here, we perform the same

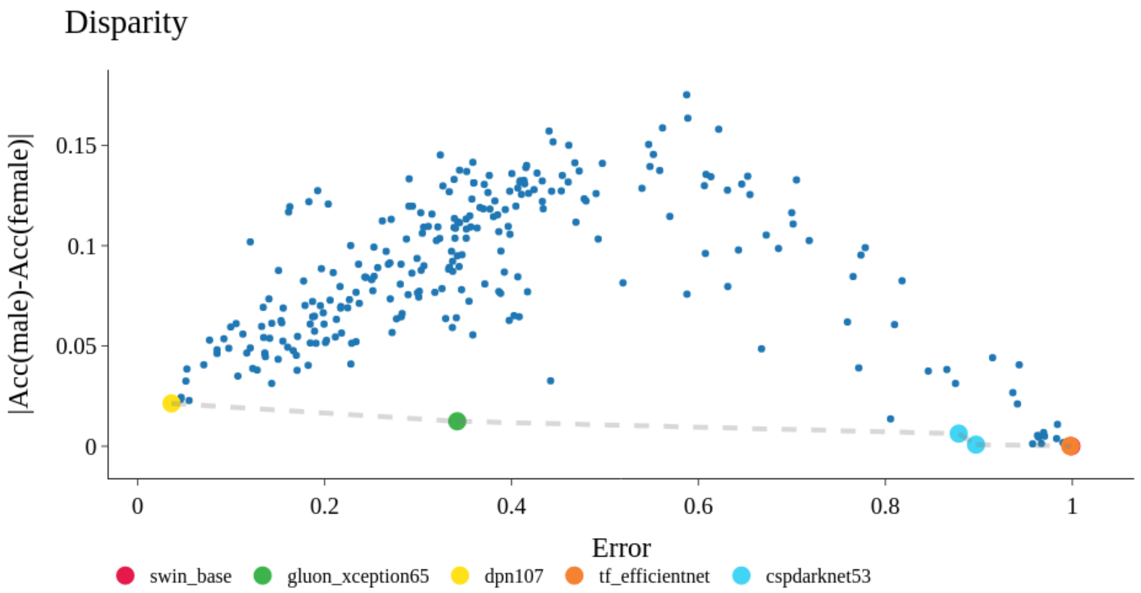


Figure 4.10: Replication of 4.7 on the CelebA validation dataset with the Disparity in accuracy metric.

experiments but evaluate on the four other metrics discussed in the face identification literature: Disparity, Rank Ratio, Ratio, and Error Ratio.

Recall, we take top performing, Pareto-optimal models from Section 4.4 and apply the three bias mitigation techniques: `Flipped`, `Angular`, and `SensitiveNets`. We also apply these same techniques to the novel architectures that we found. We report results in 4.6.

In Table 4.4, we see that in every metric, the SMAC_301 architecture is Pareto-dominant and that the SMAC_301, demonstrating the robustness of our approach.

4.5.5 Novel Architectures Generalize to Other Datasets

We also see that when we transfer our novel architecture's performance to other fairness-related dataests in facial recognition, we outperform the other architectures significantly. We take the state-of-the-art models from our experiments and test the weights from training on CelebA

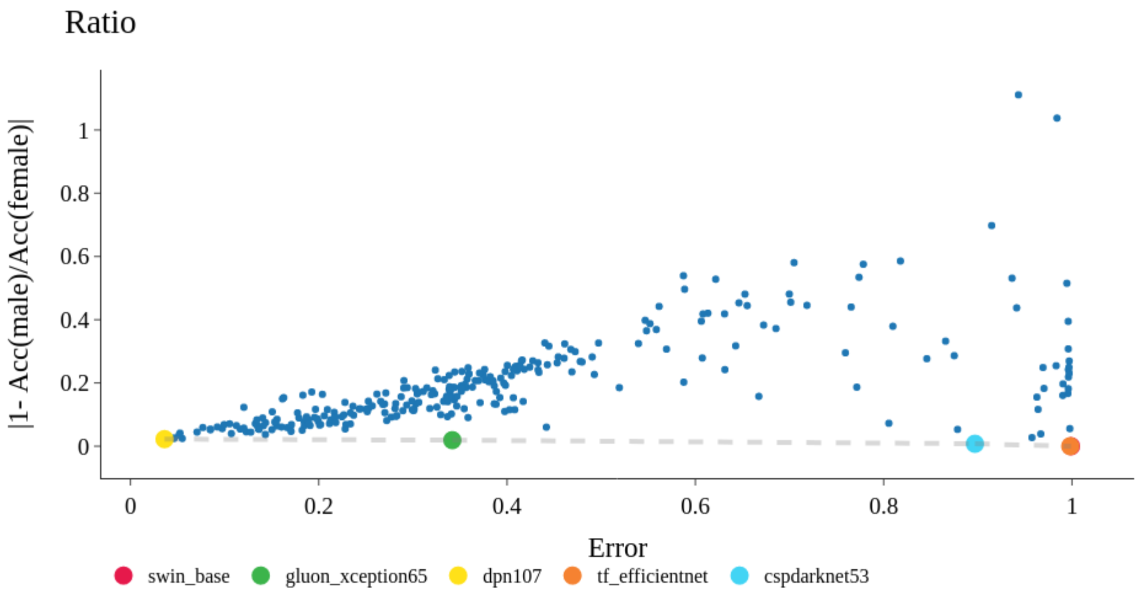


Figure 4.11: Replication of 4.7 on the CelebA validation dataset with the Ratio in accuracy metric.

and VGGFace2 on different datasets which the models did not see during training. Specificaly, we transfer the evaluation of the trained model weights from CelebA and VGGFace2 onto the following datasets: LFW (Huang et al., 2008), CFP_FF (Sengupta et al., 2016), CFP_FP (Sengupta et al., 2016), AgeDB (Moschoglou et al., 2017), CALFW (Zheng et al., 2017), CPLPW (Zheng and Deng, 2018). We see in Table 4.7 that our approach yields the architectures with the highest performance on the other datasets, meaning our approach is the most generalizable compared to state of the art recognition models in transfer learning to other face datasets.

4.5.6 Novel Architectures Generalize to Other Sensitive Attributes

The superiority of our novel architectures even goes beyond accuracy when transfering to other datasets — our novel architectures have superior fairness property compared to the existing architectures **even on datasets which have completely different protected attributes than were used in the architecture search**. To inspect the generalizability of our approach to other protected

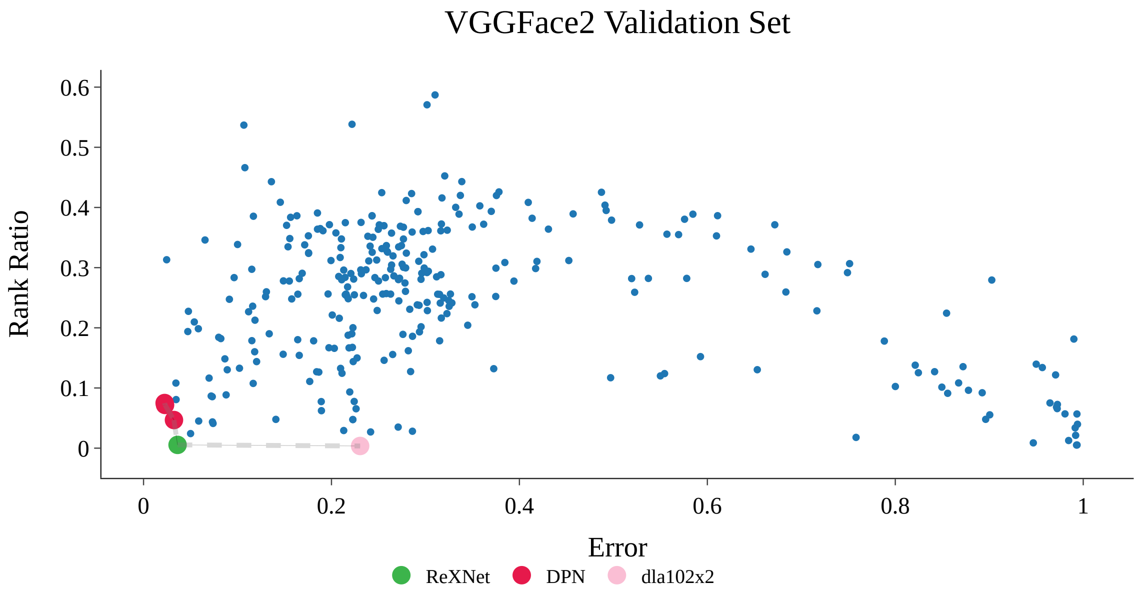


Figure 4.12: Replication of 4.8 on the VGGFace2 validation dataset with Ratio of Ranks metric.

attributes, we transferred our models pre-trained on CelebA and VGGFace2 (which have a gender presentation category) to the RFW dataset (Wang et al., 2019a) which includes a protected attribute for race. We see that our novel architectures always outperforms the existing architectures across all five fairness metrics studied in this work. See Table 4.8 for more details on each metric. They are always on the Pareto front for all fairness metrics considered, and mostly Pareto-dominate all other architectures on this task. In this setting, since the race label in RFW is not binary, the Rank Disparity metric considered in Table 4.8.

4.5.7 Novel Architectures Have Less Linear-Separability of Protected Attributes

Our comprehensive evaluation of multiple face recognition benchmarks establishes the importance of architectures for fairness in face-recognition. However, it is natural to question what makes the discovered architectures fair in the first place? To answer this question we use linear probing to dissect the intermediate features of our searched architectures and DPNs, which

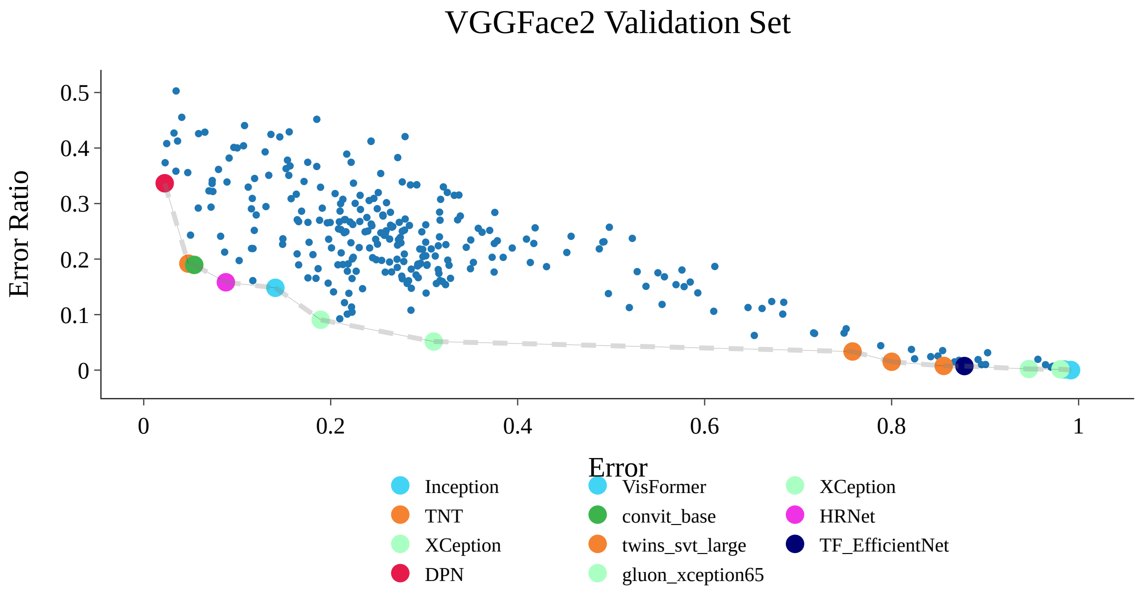


Figure 4.13: Replication of 4.8 on the VGGFace2 validation dataset with Ratio of Errors metric.

our search space is base upon. Intuitively given that our networks are trained only on the task of face recognition, we do not want the intermediate feature representations to implicitly exploit knowledge about protected attributes (eg: gender). To this end we insert linear probes (Alain and Bengio, 2016) at the last two layers of different pareto-optimal DPNs and the model obtained by Neural Architecture Search. Specifically we train an MLP on the feature representations extracted from the pre-trained models and the protected attributes as labels and compute the gender-classification accuracy on a held-out set. Our simple linear probe is represented in the equation below. We consider only the last two layers, so k assumes the values of N and $N-1$ with N being the number of layers in DPNs (and the searched models). We represent the classification probabilities for the genders by gp_k:

$$gp_k = softmax(W_k + b) \tag{4.2}$$

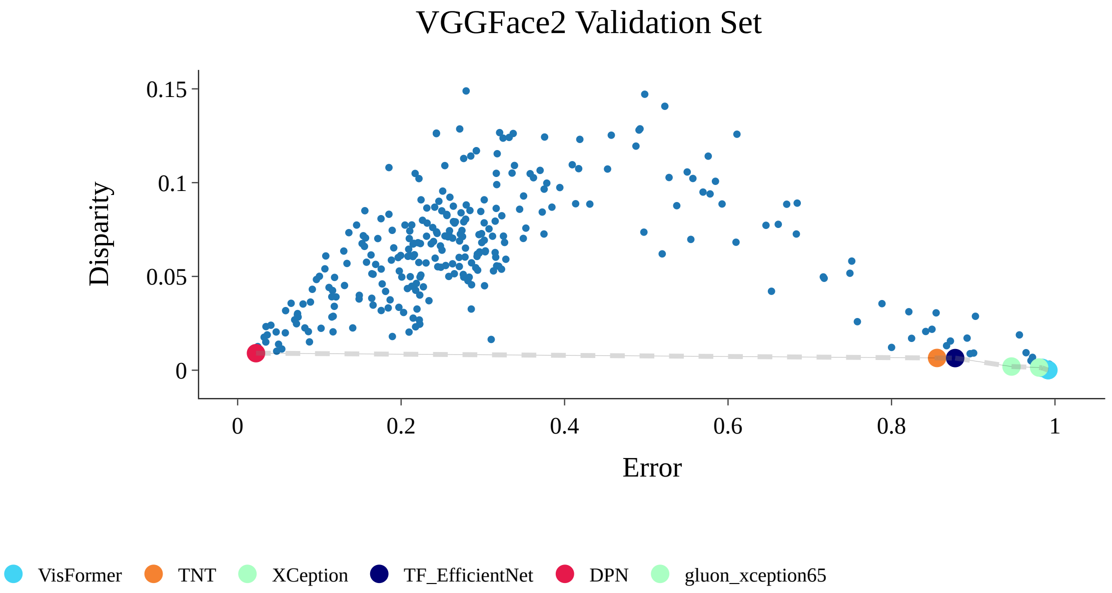


Figure 4.14: Replication of 4.8 on the VGGFace2 validation dataset with the Disparity in accuracy metric.

We provide the classification accuracies for the different pre-trained models on VGGFace2 in 4.9. It is interesting to see that the searched architectures maintain a lower classification accuracy as desirable. Inline with this observation the TSNE plots in 4.17 the DPN dislays a higher degree of separability of features.

4.6 Conclusion, Future Work and Limitations

While other bias mitigation strategies operate by changing loss functions or model parameters, we propose an entirely new direction: change the topology of the network. With our bias mitigation technique centering around Neural Architecture Search and Hyperparameter Optimization, we showed the competitiveness of our approach compared to other common bias mitigation techniques in facial recognition. We conducted the first large-scale analysis of the relationship among hyperparameters and architectural properties, and accuracy, bias, and disparity

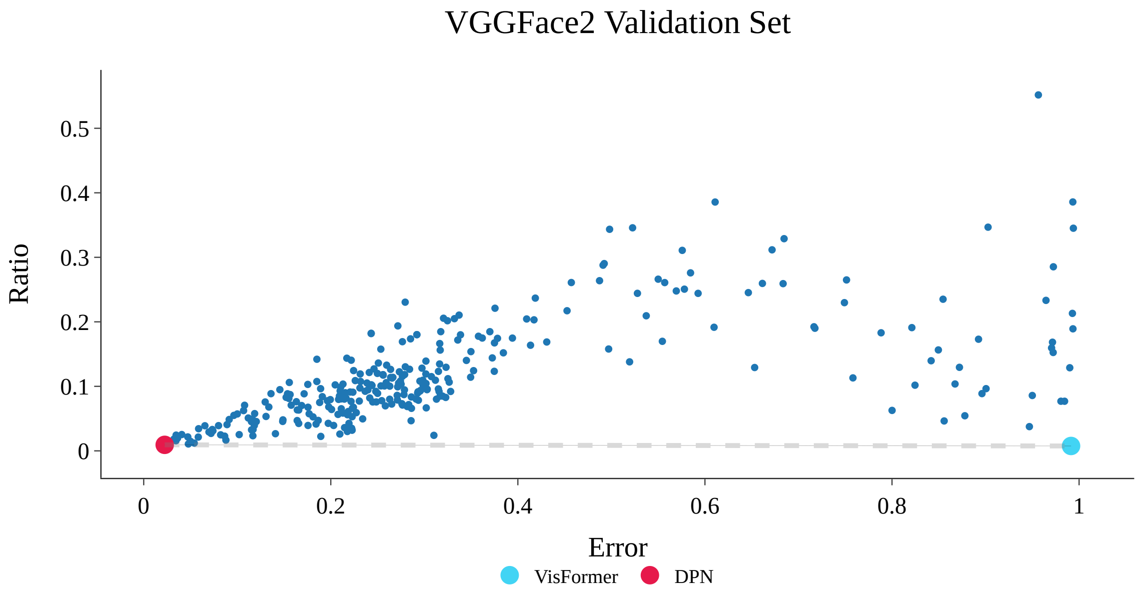


Figure 4.15: Replication of 4.8 on the VGGFace2 validation dataset with the Ratio in accuracy metric.

in predictions. We trained a set of 29 architectures totalling 355 models and 88 493 GPU hours across different loss functions and architecture heads on CelebA and VGGFace2 face identification datasets, analyzing their inductive biases for fairness and accuracy. Our observations challenge conventional practice and show that it is actually better to just search for a more fair architecture than it is to adjust an unfair one. The implicit convention of choosing the highest-accuracy architectures is a sub-optimal strategy, and we suggest that architectures and hyperparameters play a significant role in determining the fairness-accuracy tradeoff.

Since our work lays the foundation for studying NAS+HPO for fairness, it opens up a plethora of opportunities for future work. We expect the future work in this direction to focus on studying different multi-objective algorithms (Fu and Liu, 2019; Laumanns and Ocenasek, 2002) and NAS techniques (Liu et al., 2018a; Zela et al., 2019; White et al., 2021) to search for inherently fairer models. Further, it would be interesting to study how the properties of the architectures discovered translate across different demographics and populations. Another potential direction of

Table 4.4: Comparison bias mitigation techniques where the SMAC models were found on VGGFace2 with NAS bias mitigation technique and the other three techniques are standard in facial recognition: Flipped (Chang et al., 2020), Angular (Morales et al., 2020), and Discriminator (Wang and Deng, 2020). Items in bold are Pareto-optimal. The values show (Error;*metric).*

		Rank Disparity				Disparity		
Model	Baseline	Flipped	Angular	SensitiveNets	Baseline	Flipped	Angular	SensitiveNets
SMAC_301	**(3.66;0.23)**	**(4.95;0.18)**	(4.14;0.25)	(6.20;0.41)	**(3.66;0.03)**	**(4.95;0.02)**	(4.14;0.04)	(6.14;0.04)
DPN	(3.56;0.27)	(5.87;0.32)	(6.06;0.36)	(4.76;0.34)	(3.98;0.04)	(5.87;0.05)	(6.06;0.05)	(4.78;0.05)
ReXNet	(4.09;0.27)	(5.73;0.45)	(5.47;0.26)	(4.75;0.25)	(4.09;0.03)	(5.73;0.05)	(5.47;0.05)	(4.75;0.04)
Swin	(5.47;0.38)	(5.75;0.44)	(5.23;0.25)	(5.03;0.30)	(5.47;0.05)	(5.75;0.04)	(5.23;0.04)	(5.03;0.04)
		Rank Ratio				Ratio		
Model	Baseline	Flipped	Angular	SensitiveNets	Baseline	Flipped	Angular	SensitiveNets
SMAC_301	**(3.66;0.37)**	**(4.95;0.21)**	(4.14;0.39)	(6.14;0.41)	**(3.66;0.03)**	**(4.95;0.02)**	(4.14;0.04)	(6.14;0.05)
DPN	(3.98;0.49)	(5.87;0.49)	(6.06;0.54)	(4.78;0.49)	(3.98;0.04)	(5.87;0.06)	(6.06;0.06)	(4.78;0.05)
ReXNet	(4.09;0.41)	(5.73;0.53)	(5.47;0.38)	(4.75;0.34)	(4.09;0.04)	(5.73;0.05)	(5.47;0.05)	(4.75;0.04)
Swin	(5.47;0.47)	(5.75;0.47)	(5.23;0.42)	(5.03;0.43)	(5.47;0.05)	(5.75;0.05)	(5.23;0.05)	(5.03;0.05)
		Error Ratio						
Model	Baseline	Flipped	Angular	SensitiveNets				
SMAC_301	**(3.66;0.58)**	**(4.95;0.29)**	(4.14;0.60)	(6.14;0.52)				
DPN	(3.98;0.65)	(5.87;0.62)	(6.06;0.62)	(4.78;0.69)				
ReXNet	(4.09;0.60)	(5.73;0.57)	(5.47;0.59)	(4.75;0.58)				
Swin	(5.47;0.60)	(5.75;0.56)	(5.23;0.60)	(5.03;0.60)				

future work is including priors and beliefs about fairness in the society from experts to further improve and aid NAS+HPO methods for fairness by integrating expert knowledge. Finally, given the societal importance, it would be interesting to study how our findings translate to real-life face recognition systems under deployment.

While our work is a step forward in both studying the relationship among architectures, hyperparameters, and bias, and in using NAS techniques to mitigate bias in face recognition models, there are important limitations to keep in mind. Since we have studied our findings on only a few datasets, these may not generalize to other datasets and fairness metrics. Secondly, since face recognition applications span government surveillance (Hill, 2020b), target identification from drones (Marson and Forrest, 2021), and identification in personal photo repositories (Google, 2021b), our findings need to be studied thoroughly across different demographics before they

Table 4.5: Taking the highest performing models from the Pareto front of both VGGFace2 and CelebA, we transfer their evaluation onto six other common face recognition datasets: LFW (Huang et al., 2008), CFP_FF (Sengupta et al., 2016), CFP_FP (Sengupta et al., 2016), AgeDB (Moschoglou et al., 2017), CALFW (Zheng et al., 2017), CPLPW (Zheng and Deng, 2018). The novel architectures which we found with our bias mitigation strategy significantly out perform all other models.

Architecture (trained on VGGFace2)	LFW	CFP_FF	CFP_FP	AgeDB	CALFW	CPLPW
Rexnet_100	82.6	80.9142	65.514	59.1833	68.23	62.149
DPN_SGD	93.0	91.8142	78.957	71.866	78.266	72.966
DPN_AdamW	78.66	77.17	64.35	61.316	64.78	60.3
SMAC_301	**96.449**	**95.17**	**87.35**	**81.533**	**85.916**	**82.25**

Architecture (trained on CelebA)	LFW	CFP_FF	CFP_FP	AgeDB	CALFW	CPLFW
Rexnet_200	71.18	73.62	54.07	56.31	61.01	52.22
DPN_CosFace	88.86	90.47	68.53	64.84	76.09	60.66
DPN_MagFace	85.88	89.03	61.30	60.00	73.50	55.53
DenseNet161	81.72	81.88	64.82	55.16	65.7	58.40
Ese_Vovnet39	73.31	74.42	63.33	50.00	59.86	57.93
SMAC_000	**94.98**	**95.60**	74.24	80.23	84.73	64.22
SMAC_010	94.22	95.08	**75.14**	**82.35**	**85.35**	**66.26**
SMAC_680	87.45	90.34	64.22	61.28	76.16	56.16

could be deployed in real-life face recognition systems. Further, it is important to consider how the mathematical notions of fairness used in research translate to those actually impacted (Saha et al., 2020), which is a broad concept without a concise definition. Before deploying a particular system that is meant to improve fairness in a real-life application, we should always critically ask ourselves whether doing so would indeed prove beneficial to those impacted by the given sociotechnical system under consideration or whether it falls into one of the traps described by Selbst et al. (2019). In contrast to some other works, we do, however, feel, that our work helps to overcome the portability trap since it empowers domain experts to optimize for the right fairness metric, in connection with public policy experts, for the problem at hand rather than only narrowly optimizing one specific metric.

Table 4.6: Comparison bias mitigation techniques where the SMAC models were found with NAS bias mitigation technique and the other three techniques are standard in facial recognition: Flipped (Chang et al., 2020), Angular (Morales et al., 2020), and Discriminator (Wang and Deng, 2020). Items in bold are Pareto-optimal. The values show (Error;Rank Disparity).

	Trained on VGGFace2					Trained on CelebA			
Model	Baseline	Flipped	Angular	SensitiveNets	Model	Baseline	Flipped	Angular	SensitiveNets
SMAC_301	**(3.66;0.23)**	**(4.95;0.18)**	(4.14;0.25)	(6.20;0.41)	SMAC_000	(3.25;2.18)	**(5.20;0.03)**	(3.45;2.28)	(3.45;2.18)
DPN	(3.56;0.27)	(5.87;0.32)	(6.06;0.36)	(4.76;0.34)	SMAC_010	(4.14;2.27)	(12.27; 5.46)	(4.50;2.50)	(3.99;2.12)
ReXNet	(4.09;0.27)	(5.73;0.45)	(5.47;0.26)	(4.75;0.25)	SMAC_680	**(3.22;1.96)**	(12.42;4.50)	(3.80;4.16)	(3.29;2.09)
Swin	(5.47;0.38)	(5.75;0.44)	(5.23;0.25)	(5.03;0.30)	ArcFace	(11.30;4.6)	(13.56;2.70)	(9.90;5.60)	(9.10;3.00)

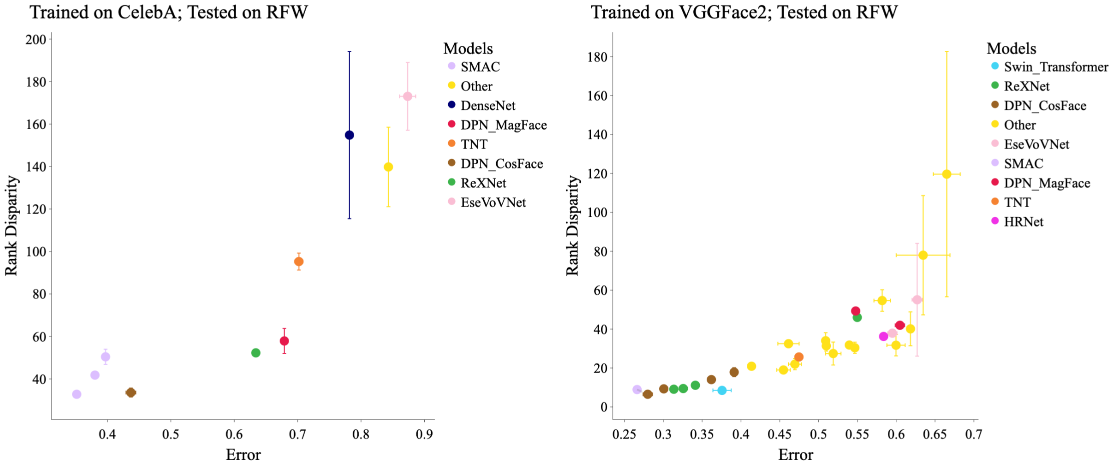


Figure 4.16: Models trained on CelebA (left) and VGGFace2 (right) evaluated on a dataset with a different protected attribute, specifically on RFW with the racial attribute, and with the Rank Disparity metric. The novel architectures out perform the existing architectures in both settings.

Table 4.7: We transfer the evaluation of top performing models on VGGFace2 and CelebA onto six other common face recognition datasets: LFW (Huang et al., 2008), CFP_FF (Sengupta et al., 2016), CFP_FP (Sengupta et al., 2016), AgeDB (Moschoglou et al., 2017), CALFW (Zheng et al., 2017), CPLPW (Zheng and Deng, 2018). The novel architectures which we found with our bias mitigation strategy significantly out perform all other models. Full table is reported in Table 4.5.

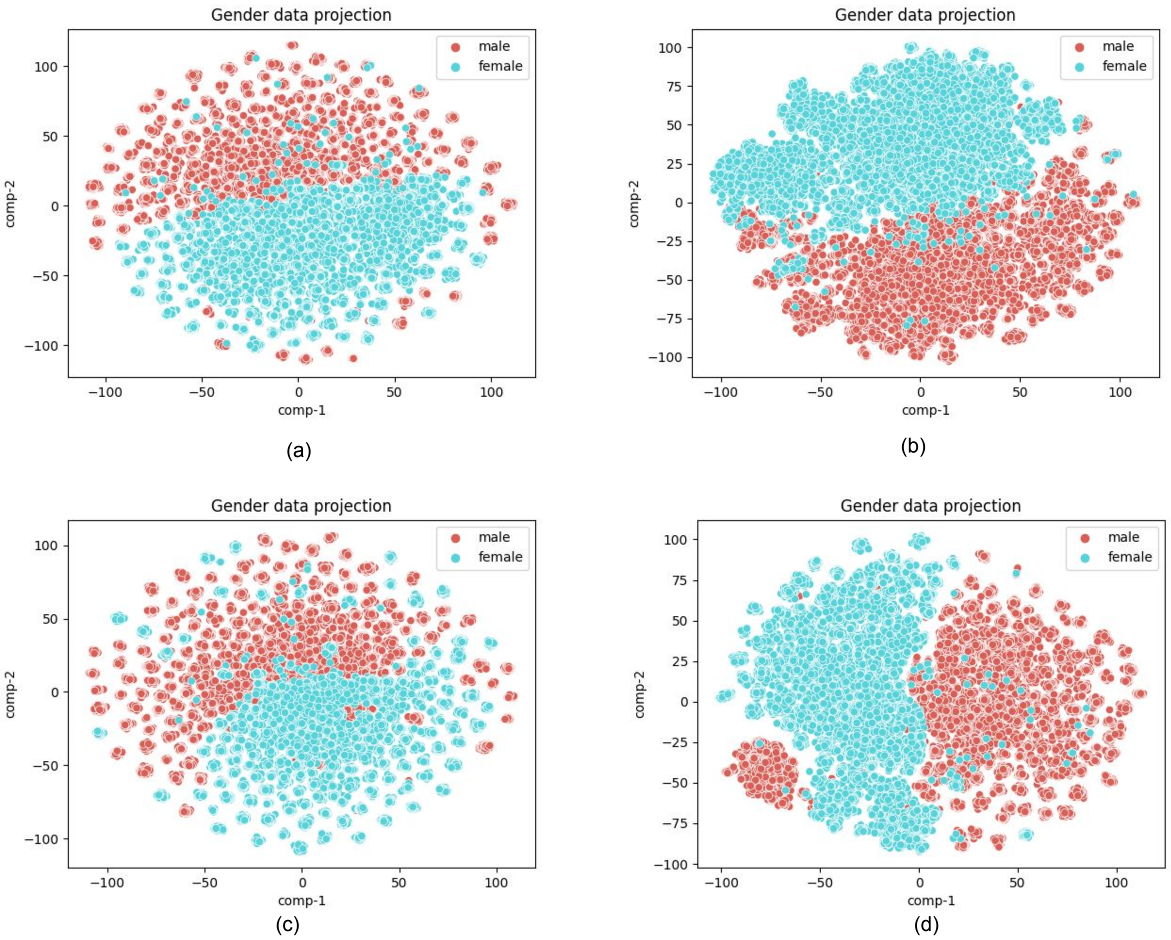
Architecture (trained on VGGFace2)	LFW	CFP_FF	CFP_FP	AgeDB	CALFW	CPLPW
Rexnet_100	82.6	80.9142	65.514	59.1833	68.23	62.149
DPN_SGD	93.0	91.8142	78.957	71.866	78.266	72.966
DPN_AdamW	78.66	77.17	64.35	61.316	64.78	60.3
SMAC_301	**96.449**	**95.17**	**87.35**	**81.533**	**85.916**	**82.25**
Architecture (trained on CelebA)	LFW	CFP_FF	CFP_FP	AgeDB	CALFW	CPLFW
DPN_CosFace	88.86	90.47	68.53	64.84	76.09	60.66
DPN_MagFace	85.88	89.03	61.30	60.00	73.50	55.53
SMAC_000	**94.98**	**95.60**	74.24	80.23	84.73	64.22
SMAC_010	94.22	95.08	**75.14**	**82.35**	**85.35**	**66.26**

Table 4.8: Taking the highest performing models from the Pareto front of both VGGFace2 and CelebA, we transfer their evaluation onto a dataset with a different protected attribue – race – on the RFW dataset (Wang et al., 2019a). The novel architectures which we found with our bias mitigation strategy are always on the Pareto front, and mostly Pareto-dominant of the traditional architectures.

Fairness Metric	Transfer from CelebA	Transfer from VGGFace2
Rank Disparity	Pareto Dominant	Pareto Optimal
Disparity	Pareto Dominant	Pareto Dominant
Rank Ratio	Pareto Optimal	Pareto Optimal
Ratio	Pareto Dominant	Pareto Dominant
Error Ratio	Pareto Optimal	Pareto Optimal

Table 4.9: Linear Probes on VGGFace2. Lower accuracy is better

Architecture (pre-trained on VGGFace2)	**Accuracy on Layer N**	**Accuracy on Layer N-1**
DPN_MagFace_SGD	86.042%	95.461%
DPN_CosFace_SGD	90.719%	93.787%
DPN_CosFace_AdamW	87.385%	94.444%
SMAC_301	**69.980%**	**68.240%**



(a)

(b)

(c)

(d)

Figure 4.17: TSNE plots for models pretrained on VGGFace2 on the test-set *(a)* SMAC model last layer *(b)* DPN MagFace on the last layer *(b)* SMAC model second last layer *(b)* DPN MagFace on the second last layer. Note the better linear separability for DPN MagFace in comparison with the SMAC model

I've learned that people will forget what you said, people will forget what you did, but people will never forget how you made them feel.

Maya Angelou

Chapter 5: Open Questions

In this chapter, I'll present some future research directions and a list of open questions that arose from the research performed as part of this thesis, or from discussions with other researchers and practioners in the area.

5.1 Implications of Adversarial Robustness Bias

The main result from Chapter 2 is that there is an interplay between where data are positioned in space and where decision boundaries are drawn. Further, that this interplay can lead to some groups of data being much closer to the decision boundary than others. In the thesis, we uncover some examples of this and demonstrate that the behavior can be seen to be robust across different datasets and models. However, an open question remain which we were unable to discuss in this research.

Open Question #1: *How does the topology of the data manifold relate to the presence of robustness bias?*

When we ask why we observed adversarial robustness in the wild in Chapter 2, one hypothesis that comes to mind is that these groups of points which are closer to the decision boundary may be less compact in either input space or feature space. The idea here being that if a group of points is less compact, then drawing a decision boundary around those points could naturally place that decision boundary closer to those points. One way to test this theory would be to look at the topological features of these points in space and see if any emergent patterns appear pertaining to, perhaps, correlation to the adversarial robustness metric.

5.2 Causal Reasoning Behind Robustness Disparities in Face Detection

In Chapter 3, we explore the disparities in the robustness of faces to natural corruptions. We saw some results that align with past work in topics of fairness in facial analysis systems; specifically we observed that individuals that were older, darker skinned, or dimly lit were more susceptible to corruptions causing missed face detections than their peers. However, we saw a surprising result that masculine-presenting individuals were those who were more susceptible. One question we can pose is why?

Open Question #2: *Why are masculine-presenting individuals more likely to be missed by a face detection system after a natural image corruption than their feminine-presenting counterparts?*

One hypothesis that might be at play here is the size of the face in the image. In the analysis above, we did not control for bounding box size of the face. That variable could be added to the regressions which we ran in Appendix A to observe how bounding box size impacts the final analysis.

Additionally, our analysis was performed primarily within the gender binary, which can obscure the impacts of these facial analysis systems on those of us who live outside the gender binary. Thus, we could ask how the face detection softwares perform on these individuals:

Open Question #3: *What are the experiences of trans/non-binary individuals with face detection?*

Of course, a research direction which pursues this question, must be carefully considered — both in terms of it's intentions and impacts. Drawing on work concerning gender, with an eye towards trans and non-binary experiences, the operalization of gender in facial recognition has long been trans-exclusionary (Hamidi et al., 2018; Keyes, 2018). While some work like the CCD dataset (Hazirbas et al., 2021) does ask for actor supplied gender labels with an open ability to provide their own text, the number of individuals who fall outside the binary in these datasets are scant. The community as a whole needs to do better to support the experiences of trans and non-binary individuals.

5.3 Applicability of NAS+HPO to Other Domains

We saw in Chapter 4 that the bias mitigation approach which we described can be vastly beneficial in the domain of face identification. However, future work should be conducted to see whether this approach is beneficial in other domains.

Open Question #4: *Does the NAS+HPO bias mitigation technique work in other vision tasks like chest x-rays or visual domains with limited data?*

First, I think it'd be easiest to apply our techniques, proven to be superior on face identification, to other visual data domains. Another visual domain which could take an easy transferal of the approach would be the chest X-ray dataset CheXpert (Irvin et al., 2019). This ask would be an easily application of our existing method and codebase to this different dataset.

Another straightforward application of the approach would be to other vision-based domains with fairness imposed via imbalanced data classes. As is often common in many commercial applications, there may be severely limited number of examples of data which are of interest to a company, but performance on those types of data are important. This can be cast under a fairness problem as well where performance on the class with fewer examples should be preserved. Types of this exploration can be found in the works like that of Buda et al. (2018) and Wang et al. (2020b).

Additionally, we can think of other data modalities beyond visual data which could benefit from NAS+HPO bias mitigation strategies. Specifically, we ask:

Open Question #5: *How do bias mitigation strategies in tabular data domains compare to the NAS+HPO bias mitigation presented in this work?*

We know well that most fairness research and literature in bias mitigation exist within the realm of tabular data. Famous tabular datasets exist and are well-studied Becker and Kohavi (1996); Dieterich et al. (2016), and critiqued (Ding et al., 2021; Bao et al., 2021) in the fairness literature. However, this thesis did not examine tabular data at all.

As deep learning for tabular data continues to develop, there is the opportunity to combine both finding good neural architectures for tabular data while simultaneously addressing bias concerns within those architectures and their outputs. I'm excited by the potential confluence of these two areas and hope to continue to study where and how NAS+HPO bias mitigations can be combined with tabular architectural and hyperparameter searches.

Appendix A: Additional Results: Robustness Bias in Face Detection

A.1 Statistical Significance Regressions for Average Precision

A.1.1 Main Tables

AP p-values for pairwise Wilcoxon test with Bonferroni correction for model on the Adience dataset can befound in Table A.2

AP p-values for pairwise Wilcoxon test with Bonferroni correction for model on the CCD dataset can befound in Table A.3

AP p-values for pairwise Wilcoxon test with Bonferroni correction for model on the MIAP dataset can befound in Table A.4

AP p-values for pairwise Wilcoxon test with Bonferroni correction for model on the UTK dataset can befound in Table A.5

A.1.2 AP — Corruption Comparison Claims

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on AWS and Adience can be found in Table A.6

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on Azure and Adience can be found in Table A.7

Table A.1: The best and worst performing perturbations for each dataset and model.

		AWS	Azure	GCP	MogFace	TinaFace	Yolo5Face
Adience	Best	Brightness	Brightness	Pixelate	Brightness	Brightness	Brightness
	Worst	Impulse Noise	Shot Noise	Snow	Impulse Noise	Impulse Noise	Impulse Noise
CCD	Best	Glass Blur	Glass Blur	Glass Blur	Glass Blur	Glass Blur	Glass Blur
	Worst	Zoom Blur	Zoom Blur	Zoom Blur	Zoom Blur	Zoom Blur	Zoom Blur
MIAP	Best	Brightness	Glass Blur	JPEG Compression	Brightness	Brightness	Brightness
	Worst	Zoom Blur	Zoom Blur	Zoom Blur	Zoom Blur	Zoom Blur	Zoom Blur
UTKFace	Best	Brightness	Brightness	JPEG Compression	Brightness	Brightness	Brightness
	Worst	Elastic Transform	Elastic Transform	Elastic Transform	Elastic Transform	Impulse Noise	Shot Noise

Table A.2: AP. Pairwise Wilcoxon test with Bonferroni correction for model on Adience

	AWS	Azure	GCP	MogFace	TinaFace
Azure	0				
GCP	0	0			
MogFace	0	0	0		
TinaFace	0	0	0	0	
Yolov5	0	0	0	0	0

Table A.3: AP. Pairwise Wilcoxon test with Bonferroni correction for model on CCD

	AWS	Azure	GCP	MogFace	TinaFace
Azure	0				
GCP	0	0			
MogFace	0	0.071	0		
TinaFace	0	0	0	0	
Yolov5	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on GCP and Adience can be found in Table A.8

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on Mog-

Table A.4: AP. Pairwise Wilcoxon test with Bonferroni correction for model on MIAP

	AWS	Azure	GCP	MogFace	TinaFace
Azure	0				
GCP	0	0			
MogFace	0	0	0		
TinaFace	0	0	0	0	
Yolov5	0	0	0	0	0

Table A.5: AP. Pairwise Wilcoxon test with Bonferroni correction for model on UTK

	AWS	Azure	GCP	MogFace	TinaFace
Azure	0				
GCP	0	0			
MogFace	0	0	0		
TinaFace	0	0	0	0	
Yolov5	0	0	0	0	0

Table A.6: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on AWS and Adience

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0.099													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0.779									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0.00000	0	0	0	0	0	0	0

Table A.7: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on Azure and Adience

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0.958	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0.0003						
fog	0	0	0	0.008	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table A.8: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on GCP and Adience

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0.0005												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0.278	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Face and Adience can be found in Table A.9

Table A.9: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on MogFace and Adience

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0.0001	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0.120	0			
elastic-transform	0	0	0	0	0	0	0.034	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on TinaFace and Adience can be found in Table A.10

Table A.10: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on TinaFace and Adience

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0.00001													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0.0001	0	0	0	
jpeg-compression	0	0	0	0	0	0.047	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on Yolov5 and Adience can be found in Table A.11

Table A.11: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on Yolov5 and Adience

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0.004													
impulse-noise	0	0.00000												
defocus-blur	0	0	0											
glass-blur	0	0	0	0.005										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0.00000	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0.643		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on AWS and CCD can be found in Table A.12

Table A.12: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on AWS and CCD

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00001

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on Azure and CCD can be found in Table A.13

Table A.13: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on Azure and CCD

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0.00000	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on GCP and CCD can be found in Table A.14

Table A.14: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on GCP and CCD

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0.00000	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on MogFace and CCD can be found in Table A.15

Table A.15: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on MogFace and CCD

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0.012	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on TinaFace and CCD can be found in Table A.16

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on Yolov5 and CCD can be found in Table A.17

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on AWS and MIAP can be found in Table A.18

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on Azure and MIAP can be found in Table A.19

Table A.16: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on TinaFace and CCD

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0.00000	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0.016	0	0.065	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00000

Table A.17: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on Yolov5 and CCD

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0.822

Table A.18: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on AWS and MIAP

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0.018													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0.00000	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0.009	0	0	0	0

Table A.19: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on Azure and MIAP

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0.211													
impulse-noise	0.913	0.170												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0.001	0	0.00000						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0.203	0.730	0.061	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0.068

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on GCP and MIAP can be found in Table A.20

Table A.20: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on GCP and MIAP

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0.123													
impulse-noise	0.131	0.963												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0.018	0.450	0.309	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0.006						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0.001	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0.492	0	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on MogFace and MIAP can be found in Table A.21

Table A.21: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on MogFace and MIAP

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0.247													
impulse-noise	0.024	0.001												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0.575	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on TinaFace and MIAP can be found in Table A.22

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on Yolov5 and MIAP can be found in Table A.23

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on AWS and UTK can be found in Table A.24

Table A.22: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on TinaFace and MIAP

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0.571													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0.215	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0.0004	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table A.23: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on Yolov5 and MIAP

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0.002													
impulse-noise	0.00000	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0.013	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0.00003	0	0			
elastic-transform	0.014	0	0.007	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0.014	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table A.24: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on AWS and UTK

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0.181	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0.756

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on Azure and UTK can be found in Table A.25

Table A.25: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on Azure and UTK

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0.272	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0.272						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0.084	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on GCP and UTK can be found in Table A.26

Table A.26: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on GCP and UTK

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0.003										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0.357	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0.001	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on MogFace and UTK can be found in Table A.27

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on TinaFace and UTK can be found in Table A.28

AP p-values for pairwise Wilcoxon test with Bonferroni correction for corruption on Yolov5 and UTK can be found in Table A.29

Table A.27: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on MogFace and UTK

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table A.28: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on TinaFace and UTK

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0.031	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table A.29: AP. Pairwise Wilcoxon test with Bonferroni correction for corruption on Yolov5 and UTK

	gaussian-noise	shot-noise	impulse-noise	defocus-blur	glass-blur	motion-blur	zoom-blur	snow	frost	fog	brightness	contrast	elastic-transform	pixelate
shot-noise	0													
impulse-noise	0	0												
defocus-blur	0	0	0											
glass-blur	0	0	0	0										
motion-blur	0	0	0	0	0									
zoom-blur	0	0	0	0	0	0								
snow	0	0	0	0	0	0	0							
frost	0	0	0	0	0	0	0	0						
fog	0	0	0	0	0	0	0	0	0					
brightness	0	0	0	0	0	0	0	0	0	0				
contrast	0	0	0	0	0	0	0	0	0	0	0			
elastic-transform	0	0	0	0	0	0	0	0	0	0	0	0		
pixelate	0	0	0	0	0	0	0	0	0	0	0	0	0	
jpeg-compression	0	0	0	0	0	0	0	0	0	0	0	0.010	0	0

A.1.3 AP — Age Comparison Claims

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on AWS and Adience can be found in Table A.30

Table A.30: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on AWS and Adience

	0-2	3-7	8-14	15-24	25-35	36-45	46-59
3-7	0						
8-14	0	0					
15-24	0	0	0				
25-35	0	0	0	0			
36-45	0	0	0	0	0		
46-59	0	0.00000	0	0	0	0	
60+	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on Azure and Adience can be found in Table A.31

Table A.31: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on Azure and Adience

	0-2	3-7	8-14	15-24	25-35	36-45	46-59
3-7	0						
8-14	0	0					
15-24	0	0	0				
25-35	0	0	0	0			
36-45	0	0.00000	0	0	0.00000		
46-59	0	0.118	0	0	0	0	
60+	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on GCP and Adience can be found in Table A.32

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on MogFace and Adience can be found in Table A.33

Table A.32: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on GCP and Adience

	0-2	3-7	8-14	15-24	25-35	36-45	46-59
3-7	0						
8-14	0	0					
15-24	0	0.00004	0				
25-35	0	0	0	0.134			
36-45	0.008	0	0	0	0		
46-59	0	0	0	0	0	0	
60+	0	0	0	0	0	0	0.003

Table A.33: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on MogFace and Adience

	0-2	3-7	8-14	15-24	25-35	36-45	46-59
3-7	0						
8-14	0	0					
15-24	0	0	0.945				
25-35	0	0	0.001	0.003			
36-45	0	0	0	0	0		
46-59	0	0.524	0	0	0	0	
60+	0.198	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on TinaFace and Adience can be found in Table A.34

Table A.34: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on TinaFace and Adience

	0-2	3-7	8-14	15-24	25-35	36-45	46-59
3-7	0						
8-14	0	0					
15-24	0	0	0				
25-35	0	0	0	0			
36-45	0	0	0.0001	0	0		
46-59	0	0	0.005	0	0	0	
60+	0.010	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on Yolov5 and Adience can be found in Table A.35

Table A.35: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on Yolov5 and Adience

	0-2	3-7	8-14	15-24	25-35	36-45	46-59
3-7	0.226						
8-14	0	0					
15-24	0	0.00000	0.00000				
25-35	0.00000	0.0001	0	0.049			
36-45	0	0	0	0	0		
46-59	0	0	0.0002	0	0	0.0001	
60+	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on AWS and CCD can be found in Table A.36

Table A.36: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on AWS and CCD

	19-45	45-64
45-64	0	
65+	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on Azure and CCD can be found in Table A.37

Table A.37: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on Azure and CCD

	19-45	45-64
45-64	0	
65+	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on GCP and CCD can be found in Table A.38

Table A.38: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on GCP and CCD

	19-45	45-64
45-64	0	
65+	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on MogFace and CCD can be found in Table A.39

Table A.39: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on MogFace and CCD

	19-45	45-64
45-64	0	
65+	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on TinaFace and CCD can be found in Table A.40

Table A.40: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on TinaFace and CCD

	19-45	45-64
45-64	0	
65+	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on Yolov5 and CCD can be found in Table A.41

Table A.41: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on Yolov5 and CCD

	19-45	45-64
45-64	0	
65+	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on AWS and MIAP can be found in Table A.42

Table A.42: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on AWS and MIAP

	Young	Middle	Older
Middle	0		
Older	0	0	
Unknown	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on Azure and MIAP can be found in Table A.43

Table A.43: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on Azure and MIAP

	Young	Middle	Older
Middle	0		
Older	0	0	
Unknown	0	0.00000	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on GCP and MIAP can be found in Table A.44

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on MogFace and MIAP can be found in Table A.45

Table A.44: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on GCP and MIAP

	Young	Middle	Older
Middle	0		
Older	0	0	
Unknown	0	0	0

Table A.45: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on MogFace and MIAP

	Young	Middle	Older
Middle	0		
Older	0	0	
Unknown	0	0.00000	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on TinaFace and MIAP can be found in Table A.46

Table A.46: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on TinaFace and MIAP

	Young	Middle	Older
Middle	0		
Older	0	0	
Unknown	0	0.001	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on Yolov5 and MIAP can be found in Table A.47

Table A.47: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on Yolov5 and MIAP

	Young	Middle	Older
Middle	0		
Older	0	0	
Unknown	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on AWS and UTK can be found in Table A.48

Table A.48: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on AWS and UTK

	0-18	19-45	45-64
19-45	0		
45-64	0	0	
65+	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on Azure and UTK can be found in Table A.49

Table A.49: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on Azure and UTK

	0-18	19-45	45-64
19-45	0		
45-64	0	0.570	
65+	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on GCP and UTK can be found in Table A.50

Table A.50: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on GCP and UTK

	0-18	19-45	45-64
19-45	0		
45-64	0	0	
65+	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on MogFace and UTK can be found in Table A.51

Table A.51: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on MogFace and UTK

	0-18	19-45	45-64
19-45	0		
45-64	0	0	
65+	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on TinaFace and UTK can be found in Table A.52

Table A.52: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on TinaFace and UTK

	0-18	19-45	45-64
19-45	0		
45-64	0.00000	0	
65+	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Age on Yolov5 and UTK can be found in Table A.53

Table A.53: AP. Pairwise Wilcoxon test with Bonferroni correction for Age on Yolov5 and UTK

	0-18	19-45	45-64
19-45	0		
45-64	0	0	
65+	0	0	0

A.1.4 AP — Gender Comparison Claims

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on AWS and Adience can be found in Table A.54

Table A.54: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on AWS and Adience

	Female
Male	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on Azure and Adience can be found in Table A.55

Table A.55: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on Azure and Adience

	Female
Male	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on GCP and Adience can be found in Table A.56

Table A.56: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on GCP and Adience

	Female
Male	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on MogFace and Adience can be found in Table A.57

Table A.57: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on MogFace and Adience

	Female
Male	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on TinaFace and Adience can be found in Table A.58

Table A.58: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on TinaFace and Adience

	Female
Male	0.203

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on Yolov5 and Adience can be found in Table A.59

Table A.59: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on Yolov5 and Adience

	Female
Male	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on AWS and CCD can be found in Table A.60

Table A.60: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on AWS and CCD

	Female	Male
Male	0	
Other	0.680	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on Azure and CCD can be found in Table A.61

Table A.61: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on Azure and CCD

	Female	Male
Male	0	
Other	0.171	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on GCP and CCD can be found in Table A.62

Table A.62: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on GCP and CCD

	Female	Male
Male	0	
Other	0.003	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on MogFace and CCD can be found in Table A.63

Table A.63: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on MogFace and CCD

	Female	Male
Male	0	
Other	0.806	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on TinaFace and CCD can be found in Table A.64

Table A.64: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on TinaFace and CCD

	Female	Male
Male	0	
Other	0.740	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on Yolov5 and CCD can be found in Table A.65

Table A.65: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on Yolov5 and CCD

	Female	Male
Male	0	
Other	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on AWS and MIAP can be found in Table A.66

Table A.66: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on AWS and MIAP

	Predominantly Feminine	Predominantly Masculine
Predominantly Masculine	0	
Unknown	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on Azure and MIAP can be found in Table A.67

Table A.67: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on Azure and MIAP

	Predominantly Feminine	Predominantly Masculine
Predominantly Masculine	0	
Unknown	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on GCP and MIAP can be found in Table A.68

Table A.68: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on GCP and MIAP

	Predominantly Feminine	Predominantly Masculine
Predominantly Masculine	0	
Unknown	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on MogFace and MIAP can be found in Table A.69

Table A.69: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on MogFace and MIAP

	Predominantly Feminine	Predominantly Masculine
Predominantly Masculine	0	
Unknown	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on TinaFace and MIAP can be found in Table A.70

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on Yolov5 and MIAP can be found in Table A.71

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on AWS and UTK can be found in Table A.72

Table A.70: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on TinaFace and MIAP

	Predominantly Feminine	Predominantly Masculine
Predominantly Masculine	0	
Unknown	0	0

Table A.71: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on Yolov5 and MIAP

	Predominantly Feminine	Predominantly Masculine
Predominantly Masculine	0	
Unknown	0	0

Table A.72: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on AWS and UTK

	Female
Male	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on Azure and UTK can be found in Table A.73

Table A.73: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on Azure and UTK

	Female
Male	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on GCP and UTK can be found in Table A.74

Table A.74: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on GCP and UTK

	Female
Male	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on MogFace and UTK can be found in Table A.75

Table A.75: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on MogFace and UTK

	Female
Male	0.0001

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on TinaFace and UTK can be found in Table A.76

Table A.76: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on TinaFace and UTK

	Female
Male	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Gender on Yolov5 and UTK can be found in Table A.77

Table A.77: AP. Pairwise Wilcoxon test with Bonferroni correction for Gender on Yolov5 and UTK

	Female
Male	0

A.1.5 AP — Skin Type Comparison Claims

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type on AWS and CCD can be found in Table A.78

Table A.78: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type on AWS and CCD

	Light Fitz
Dark Fitz	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type on Azure and CCD can be found in Table A.79

Table A.79: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type on Azure and CCD

	Light Fitz
Dark Fitz	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type on GCP and CCD can be found in Table A.80

Table A.80: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type on GCP and CCD

	Light Fitz
Dark Fitz	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type on MogFace and CCD can be found in Table A.81

Table A.81: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type on MogFace and CCD

	Light Fitz
Dark Fitz	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type on TinaFace and CCD can be found in Table A.82

Table A.82: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type on TinaFace and CCD

	Light Fitz
Dark Fitz	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type on Yolov5 and CCD can be found in Table A.83

Table A.83: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type on Yolov5 and CCD

	Light Fitz
Dark Fitz	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on AWS and CCD can be found in Table A.84

Table A.84: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on AWS and CCD

	Dark Fitz+Bright	Dark Fitz+Dim	Light Fitz+Bright
Dark Fitz+Dim	0		
Light Fitz+Bright	0	0	
Light Fitz+Dim	0	0.567	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on Azure and CCD can be found in Table A.85

Table A.85: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on Azure and CCD

	Dark Fitz+Bright	Dark Fitz+Dim	Light Fitz+Bright
Dark Fitz+Dim	0		
Light Fitz+Bright	0	0	
Light Fitz+Dim	0	0.076	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on GCP and CCD can be found in Table A.86

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on MogFace and CCD can be found in Table A.87

Table A.86: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on GCP and CCD

	Dark Fitz+Bright	Dark Fitz+Dim	Light Fitz+Bright
Dark Fitz+Dim	0		
Light Fitz+Bright	0	0	
Light Fitz+Dim	0	0	0

Table A.87: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on MogFace and CCD

	Dark Fitz+Bright	Dark Fitz+Dim	Light Fitz+Bright
Dark Fitz+Dim	0		
Light Fitz+Bright	0	0	
Light Fitz+Dim	0	0.316	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on TinaFace and CCD can be found in Table A.88

Table A.88: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on TinaFace and CCD

	Dark Fitz+Bright	Dark Fitz+Dim	Light Fitz+Bright
Dark Fitz+Dim	0		
Light Fitz+Bright	0	0	
Light Fitz+Dim	0	0.004	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on Yolov5 and CCD can be found in Table A.89

Table A.89: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Lighting on Yolov5 and CCD

	Dark Fitz+Bright	Dark Fitz+Dim	Light Fitz+Bright
Dark Fitz+Dim	0		
Light Fitz+Bright	0	0	
Light Fitz+Dim	0	0.00004	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on AWS and CCD can be found in Table A.90

Table A.90: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on AWS and CCD

	Dark Fitz + 19-45 + Female	Dark Fitz + 19-45 + Male	Dark Fitz + 19-45 + Other	Dark Fitz + 45-64 + Female	Dark Fitz + 45-64 + Male	Dark Fitz + 45-64 + Other	Dark Fitz + 65+ + Female	Dark Fitz + 65+ + Male	Dark Fitz + 65+ + Other	Light Fitz + 19-45 + Female	Light Fitz + 19-45 + Male	Light Fitz + 19-45 + Other	Light Fitz + 45-64 + Female	Light Fitz + 45-64 + Male	Light Fitz + 45-64 + Other	Light Fitz + 65+ + Female
Dark Fitz + 19-45 + Male	0															
Dark Fitz + 19-45 + Other	0.038	0.00000														
Dark Fitz + 45-64 + Female	0.002	0	0.760													
Dark Fitz + 45-64 + Male	0	0.00000	0	0												
Dark Fitz + 45-64 + Other	0.061	0.763	0.004	0.002	0.016											
Dark Fitz + 65+ + Female	0.00000	0.125	0.00000	0	0.050	0.263										
Dark Fitz + 65+ + Male	0	0.002	0	0	0.934	0.029	0.125									
Dark Fitz + 65+ + Other	0	0	0	0	0	0	0	0								
Light Fitz + 19-45 + Female	0	0	0	0	0	0	0	0	0							
Light Fitz + 19-45 + Male	0	0	0	0	0	0	0	0	0	0.021						
Light Fitz + 19-45 + Other	0	0	0.00005	0.00000	0	0	0	0	0	0.101	0.006					
Light Fitz + 45-64 + Female	0.006	0	0.669	0.792	0	0.003	0	0	0	0	0	0.00000				
Light Fitz + 45-64 + Male	0.003	0.016	0.0003	0.00000	0	0.530	0.002	0.00001	0	0	0	0	0.00000			
Light Fitz + 45-64 + Other	0.006	0.0001	0.067	0.042	0.00000	0.001	0.00001	0.00000	0	0.212	0.070	0.697	0.037	0.001		
Light Fitz + 65+ + Female	0.00000	0.197	0.00000	0	0.012	0.341	0.769	0.063	0	0	0	0	0	0.003	0.00001	
Light Fitz + 65+ + Male	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on Azure and CCD can be found in Table A.91

Table A.91: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on Azure and CCD

	Dark Fitz + 19-45 + Female	Dark Fitz + 19-45 + Male	Dark Fitz + 19-45 + Other	Dark Fitz + 45-64 + Female	Dark Fitz + 45-64 + Male	Dark Fitz + 45-64 + Other	Dark Fitz + 65+ + Female	Dark Fitz + 65+ + Male	Dark Fitz + 65+ + Other	Light Fitz + 19-45 + Female	Light Fitz + 19-45 + Male	Light Fitz + 19-45 + Other	Light Fitz + 45-64 + Female	Light Fitz + 45-64 + Male	Light Fitz + 45-64 + Other	Light Fitz + 65+ + Female
Dark Fitz + 19-45 + Male	0															
Dark Fitz + 19-45 + Other	0.947	0														
Dark Fitz + 45-64 + Female	0.136	0	0.389													
Dark Fitz + 45-64 + Male	0	0.00000	0	0												
Dark Fitz + 45-64 + Other	0.081	0.0002	0.116	0.236	0.00000											
Dark Fitz + 65+ + Female	0.00000	0.00002	0.0002	0.00003	0	0.236										
Dark Fitz + 65+ + Male	0	0.052	0	0	0.516	0.00001	0.00000									
Dark Fitz + 65+ + Other	0	0	0	0	0	0	0	0								
Light Fitz + 19-45 + Female	0	0	0	0	0	0	0	0	0							
Light Fitz + 19-45 + Male	0	0	0	0	0	0	0	0	0	0.012						
Light Fitz + 19-45 + Other	0.082	0	0.162	0.016	0	0.009	0.00000	0	0	0	0.00000					
Light Fitz + 45-64 + Female	0.902	0	0.990	0.143	0	0.078	0.00000	0	0	0	0	0.098				
Light Fitz + 45-64 + Male	0	0	0.00001	0	0	0.202	0.947	0.00000	0	0	0	0.00000	0			
Light Fitz + 45-64 + Other	0	0	0	0	0	0	0	0	0	0.023	0.004	0.00000	0	0		
Light Fitz + 65+ + Female	0	0.141	0	0	0.00000	0.007	0.022	0.007	0	0	0	0	0	0.007	0	
Light Fitz + 65+ + Male	0	0.082	0	0	0.104	0.00001	0.00000	0.589	0	0	0	0	0	0	0	0.010

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on GCP and CCD can be found in Table A.92

Table A.92: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on GCP and CCD

	Dark Fitz + 19-45 + Female	Dark Fitz + 19-45 + Male	Dark Fitz + 19-45 + Other	Dark Fitz + 45-64 + Female	Dark Fitz + 45-64 + Male	Dark Fitz + 45-64 + Other	Dark Fitz + 65+ + Female	Dark Fitz + 65+ + Male	Dark Fitz + 65+ + Other	Light Fitz + 19-45 + Female	Light Fitz + 19-45 + Male	Light Fitz + 19-45 + Other	Light Fitz + 45-64 + Female	Light Fitz + 45-64 + Male	Light Fitz + 45-64 + Other	Light Fitz + 65+ + Female
Dark Fitz + 19-45 + Male	0															
Dark Fitz + 19-45 + Other	0.003	0														
Dark Fitz + 45-64 + Female	0.00003	0	0.553													
Dark Fitz + 45-64 + Male	0	0.00000	0	0												
Dark Fitz + 45-64 + Other	0.0004	0	0.169	0.050	0											
Dark Fitz + 65+ + Female	0	0.008	0	0	0.340	0										
Dark Fitz + 65+ + Male	0	0.528	0	0	0.0003	0	0.012									
Dark Fitz + 65+ + Other	0	0	0	0	0	0	0	0								
Light Fitz + 19-45 + Female	0	0	0	0	0	0	0	0	0							
Light Fitz + 19-45 + Male	0	0	0	0	0	0.00000	0	0	0	0						
Light Fitz + 19-45 + Other	0	0	0	0	0	0	0	0	0	0.473	0.002					
Light Fitz + 45-64 + Female	0	0	0.0002	0	0	0.281	0	0	0	0	0	0				
Light Fitz + 45-64 + Male	0.239	0	0.030	0.008	0	0.002	0	0	0	0	0	0	0			
Light Fitz + 45-64 + Other	0	0	0.00000	0.00000	0	0.0001	0	0	0	0.654	0.153	0.947	0.0003	0		
Light Fitz + 65+ + Female	0.793	0	0.008	0.002	0	0.0004	0	0	0	0	0	0	0	0.343	0	
Light Fitz + 65+ + Male	0.269	0	0.001	0.0001	0	0.0001	0	0	0	0	0	0	0	0.058	0	0.459

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on MogFace and CCD can be found in Table A.93

Table A.93: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on MogFace and CCD

	Dark Fitz + 19-45 + Female	Dark Fitz + 19-45 + Male	Dark Fitz + 19-45 + Other	Dark Fitz + 45-64 + Female	Dark Fitz + 45-64 + Male	Dark Fitz + 45-64 + Other	Dark Fitz + 65+ + Female	Dark Fitz + 65+ + Male	Dark Fitz + 65+ + Other	Light Fitz + 19-45 + Female	Light Fitz + 19-45 + Male	Light Fitz + 19-45 + Other	Light Fitz + 45-64 + Female	Light Fitz + 45-64 + Male	Light Fitz + 45-64 + Other	Light Fitz + 65+ + Female
Dark Fitz + 19-45 + Male	0															
Dark Fitz + 19-45 + Other	0.010	0														
Dark Fitz + 45-64 + Female	0.475	0	0.003													
Dark Fitz + 45-64 + Male	0	0.00001	0	0												
Dark Fitz + 45-64 + Other	0.00004	0.085	0.00000	0.0002	0.0004											
Dark Fitz + 65+ + Female	0.0002	0	0.00000	0.001	0	0.100										
Dark Fitz + 65+ + Male	0	0.0003	0	0	0.321	0.0002	0									
Dark Fitz + 65+ + Other	0	0	0	0	0	0	0	0								
Light Fitz + 19-45 + Female	0	0	0	0	0	0	0	0	0							
Light Fitz + 19-45 + Male	0	0	0.0001	0	0	0	0	0	0	0						
Light Fitz + 19-45 + Other	0	0	0.00000	0	0	0	0	0	0	0.011	0.026					
Light Fitz + 45-64 + Female	0	0	0.324	0	0	0	0	0	0	0	0.00000	0.00001				
Light Fitz + 45-64 + Male	0	0.152	0	0	0.004	0.020	0	0.005	0	0	0	0	0			
Light Fitz + 45-64 + Other	0	0	0.00000	0	0	0	0	0	0	0.279	0.001	0.030	0.00001	0		
Light Fitz + 65+ + Female	0.0003	0	0.00000	0.003	0	0.053	0.718	0	0	0	0	0	0	0	0	
Light Fitz + 65+ + Male	0	0	0	0	0	0	0	0.00000	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on TinaFace and CCD can be found in Table A.94

Table A.94: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on TinaFace and CCD

	Dark Fitz + 19-45 + Female	Dark Fitz + 19-45 + Male	Dark Fitz + 19-45 + Other	Dark Fitz + 45-64 + Female	Dark Fitz + 45-64 + Male	Dark Fitz + 45-64 + Other	Dark Fitz + 65+ + Female	Dark Fitz + 65+ + Male	Dark Fitz + 65+ + Other	Light Fitz + 19-45 + Female	Light Fitz + 19-45 + Male	Light Fitz + 19-45 + Other	Light Fitz + 45-64 + Female	Light Fitz + 45-64 + Male	Light Fitz + 45-64 + Other	Light Fitz + 65+ + Female
Dark Fitz + 19-45 + Male	0															
Dark Fitz + 19-45 + Other	0.515	0														
Dark Fitz + 45-64 + Female	0.767	0	0.657													
Dark Fitz + 45-64 + Male	0	0.00002	0	0												
Dark Fitz + 45-64 + Other	0.009	0	0.047	0.011	0											
Dark Fitz + 65+ + Female	0.00000	0.020	0.00005	0.00000	0.00000	0.00000										
Dark Fitz + 65+ + Male	0	0.001	0	0	0.389	0	0.00001									
Dark Fitz + 65+ + Other	0	0	0	0	0	0	0	0								
Light Fitz + 19-45 + Female	0	0	0	0	0	0	0	0	0							
Light Fitz + 19-45 + Male	0	0	0	0	0	0.00000	0	0	0	0						
Light Fitz + 19-45 + Other	0.0001	0	0.006	0.0002	0	0.583	0	0	0	0	0.00001					
Light Fitz + 45-64 + Female	0.0002	0	0.130	0.001	0	0.248	0	0	0	0	0	0.044				
Light Fitz + 45-64 + Male	0	0.089	0.00000	0	0.00000	0	0.300	0.00002	0	0	0	0	0			
Light Fitz + 45-64 + Other	0	0	0	0	0	0	0	0	0	0.001	0.00000	0	0	0		
Light Fitz + 65+ + Female	0.00001	0.0002	0.001	0.00002	0	0.00000	0.352	0.00000	0	0	0	0	0	0.020	0	
Light Fitz + 65+ + Male	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on Yolov5 and CCD can be found in Table A.95

Table A.95: AP. Pairwise Wilcoxon test with Bonferroni correction for Skin Type and the interaction with Age and Gender on Yolov5 and CCD

	Dark Fitz + 19-45 + Female	Dark Fitz + 19-45 + Male	Dark Fitz + 19-45 + Other	Dark Fitz + 45-64 + Female	Dark Fitz + 45-64 + Male	Dark Fitz + 45-64 + Other	Dark Fitz + 65+ + Female	Dark Fitz + 65+ + Male	Dark Fitz + 65+ + Other	Light Fitz + 19-45 + Female	Light Fitz + 19-45 + Male	Light Fitz + 19-45 + Other	Light Fitz + 45-64 + Female	Light Fitz + 45-64 + Male	Light Fitz + 45-64 + Other	Light Fitz + 65+ + Female
Dark Fitz + 19-45 + Male	0															
Dark Fitz + 19-45 + Other	0	0														
Dark Fitz + 45-64 + Female	0.881	0	0													
Dark Fitz + 45-64 + Male	0	0.933	0	0												
Dark Fitz + 45-64 + Other	0.029	0.198	0.00000	0.026	0.224											
Dark Fitz + 65+ + Female	0	0.001	0	0	0.001	0.003										
Dark Fitz + 65+ + Male	0	0	0	0	0	0	0									
Dark Fitz + 65+ + Other	0	0.00000	0	0	0.00000	0.00000	0.00000	0.142								
Light Fitz + 19-45 + Female	0	0	0.490	0	0	0	0	0	0							
Light Fitz + 19-45 + Male	0	0	0.710	0	0	0	0	0	0	0.502						
Light Fitz + 19-45 + Other	0	0	0.014	0	0	0	0	0	0	0.028	0.008					
Light Fitz + 45-64 + Female	0.006	0	0	0.006	0.00000	0.324	0	0	0	0	0	0				
Light Fitz + 45-64 + Male	0	0.050	0	0	0.067	0.625	0.00000	0	0.00000	0	0	0	0.0003			
Light Fitz + 45-64 + Other	0.00004	0	0.093	0.00004	0	0.00000	0	0	0	0.150	0.095	0.708	0.00000	0		
Light Fitz + 65+ + Female	0	0.634	0	0	0.654	0.341	0.002	0	0.00000	0	0	0	0.0003	0.408	0	
Light Fitz + 65+ + Male	0	0	0	0	0	0.00000	0.001	0	0.0001	0	0	0	0	0	0	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Lighting on AWS and CCD can be found in Table A.96

Table A.96: AP. Pairwise Wilcoxon test with Bonferroni correction for Lighting on AWS and CCD

	Bright
Dim	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Lighting on Azure and CCD can be found in Table A.97

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Lighting on GCP and CCD can be found in Table A.98

Table A.97: AP. Pairwise Wilcoxon test with Bonferroni correction for Lighting on Azure and CCD

	Bright
Dim	0

Table A.98: AP. Pairwise Wilcoxon test with Bonferroni correction for Lighting on GCP and CCD

	Bright
Dim	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Lighting on MogFace and CCD can be found in Table A.99

Table A.99: AP. Pairwise Wilcoxon test with Bonferroni correction for Lighting on MogFace and CCD

	Bright
Dim	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Lighting on TinaFace and CCD can be found in Table A.100

Table A.100: AP. Pairwise Wilcoxon test with Bonferroni correction for Lighting on TinaFace and CCD

	Bright
Dim	0

AP p-values for pairwise Wilcoxon test with Bonferroni correction for Lighting on Yolov5 and CCD can be found in Table A.101

Table A.101: AP. Pairwise Wilcoxon test with Bonferroni correction for Lighting on Yolov5 and CCD

	Bright
Dim	0

Bibliography

Adel, T., Valera, I., Ghahramani, Z., and Weller, A. (2019). One-network adversarial fairness. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 2412–2420.

Adeli, E., Zhao, Q., Pfefferbaum, A., Sullivan, E. V., Fei-Fei, L., Niebles, J. C., and Pohl, K. M. (2021). Representation learning with statistical independence to mitigate bias. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2513–2523.

Agarwal, A., Beygelzimer, A., Dudik, M., Langford, J., and Wallach, H. (2018). A reductions approach to fair classification. In *International Conference on Machine Learning*, pages 60–69.

Alain, G. and Bengio, Y. (2016). Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*.

Allyn, B. (2020). 'The computer got it wrong': How facial recognition led to false arrest of black man. *NPR, June*, 24.

Bao, M., Zhou, A., Zottola, S. A., Brubach, B., Desmarais, S., Horowitz, A. S., Lum, K., and Venkatasubramanian, S. (2021). It's compaslicated: The messy relationship between rai datasets and algorithmic fairness benchmarks. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Barocas, S., Hardt, M., and Narayanan, A. (2017). Fairness in machine learning. *NIPS Tutorial*.

Barocas, S., Hardt, M., and Narayanan, A. (2019). *Fairness and Machine Learning*. fairmlbook.org. http://www.fairmlbook.org.

Barocas, S. and Selbst, A. D. (2016). Big data's disparate impact. *California Law Review*, 104:671.

Becker, B. and Kohavi, R. (1996). Adult. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5XW20.

Bello, I., Fedus, W., Du, X., Cubuk, E. D., Srinivas, A., Lin, T.-Y., Shlens, J., and Zoph, B. (2021). Revisiting resnets: Improved training and scaling strategies. *Advances in Neural Information Processing Systems*, 34:22614–22627.

Benthall, S. and Haynes, B. D. (2019). Racial categories in machine learning. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 289–298.

Beutel, A., Chen, J., Zhao, Z., and Chi, E. H. (2017). Data decisions and theoretical implications when adversarially learning fair representations. *arXiv preprint arXiv:1707.00075*.

Beyea, A. and Kebede, M. (2021). Maine's facial recognition law shows bipartisan support for protecting privacy. *TechCrunch*.

Biega, A. J., Gummadi, K. P., and Weikum, G. (2018). Equity of attention: Amortizing individual fairness in rankings. In *ACM Conference on Research and Development in Information Retrieval (SIGIR)*, page 405?414.

Binns, R. (2017). Fairness in machine learning: Lessons from political philosophy. *Proceedings of Machine Learning Research*, 81:1–11.

Boser, B. E., Guyon, I. M., and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *Conference on Learning Theory (COLT)*, page 144?152.

Buda, M., Maki, A., and Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. *Neural networks*, 106:249–259.

Buolamwini, J. and Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81, pages 77–91.

Cai, H., Gan, C., Wang, T., Zhang, Z., and Han, S. (2019). Once-for-all: Train one network and specialize it for efficient deployment. *arXiv preprint arXiv:1908.09791*.

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

Calmon, F., Wei, D., Vinzamuri, B., Natesan Ramamurthy, K., and Varshney, K. R. (2017). Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems 30*, NIPS'17, pages 3992–4001.

Cao, Q., Shen, L., Xie, W., Parkhi, O. M., and Zisserman, A. (2018). Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 67–74. IEEE.

Carlini, N. and Wagner, D. (2017). Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (S&P)*, pages 39–57.

Chang, H., Nguyen, T. D., Murakonda, S. K., Kazemi, E., and Shokri, R. (2020). On adversarial bias and the robustness of fair machine learning. *arXiv preprint arXiv:2006.08669*.

Chen, Y., Li, J., Xiao, H., Jin, X., Yan, S., and Feng, J. (2017). Dual path networks. *Advances in neural information processing systems*, 30.

Chen, Z., Xie, L., Niu, J., Liu, X., Wei, L., and Tian, Q. (2021). Visformer: The vision-friendly transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 589–598.

Cherepanova, V., Goldblum, M., Foley, H., Duan, S., Dickerson, J. P., Taylor, G., and Goldstein, T. (2021). Lowkey: leveraging adversarial attacks to protect social media users from facial recognition. In *International Conference on Learning Representations (ICLR)*.

Cherepanova, V., Nanda, V., Goldblum, M., Dickerson, J. P., and Goldstein, T. (2023). Technical challenges for training fair neural networks. *6th AAAI/ACM Conference on AI, Ethics, and Society*.

Cherepanova, V., Reich, S., Dooley, S., Souri, H., Goldblum, M., and Goldstein, T. (2022). A deep dive into dataset imbalance and bias in face identification. *arXiv preprint arXiv:2203.08235*.

Chi, J., Walia, E., Babyn, P., Wang, J., Groot, G., and Eramian, M. (2017). Thyroid nodule classification in ultrasound images by fine-tuning deep convolutional neural network. *Journal of digital imaging*, 30(4):477–486.

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258.

Chouldechova, A. (2017). Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, 5(2):153–163.

Chouldechova, A. and Roth, A. (2018). The frontiers of fairness in machine learning. *arXiv preprint arXiv:1810.08810*.

Chu, X., Tian, Z., Wang, Y., Zhang, B., Ren, H., Wei, X., Xia, H., and Shen, C. (2021). Twins: Revisiting the design of spatial attention in vision transformers. *Advances in Neural Information Processing Systems*, 34:9355–9366.

Cohen, J. M., Rosenfeld, E., and Kolter, J. Z. (2019). Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning (ICML)*.

Cramer, H., Vaughan, J. W., Holstein, K., Wallach, H., Garcia-Gathright, J., III, H. D., Dudok, M., and Reddy, S. (2019). Challenges of incorporating algorithmic fairness into industry practice. *FAT* Tutorial*.

Crawford, K. and Paglen, T. (2019). Excavating ai: The politics of images in machine learning training sets.

Cruz, A. F., Saleiro, P., Belém, C., Soares, C., and Bizarro, P. (2020). A bandit-based algorithm for fairness-aware hyperparameter optimization. *arXiv preprint arXiv:2010.03665*.

Dai, X., Wan, A., Zhang, P., Wu, B., He, Z., Wei, Z., Chen, K., Tian, Y., Yu, M., Vajda, P., et al. (2021). Fbnetv3: Joint architecture-recipe search using predictor pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16276–16285.

d'Ascoli, S., Touvron, H., Leavitt, M., Morcos, A., Biroli, G., and Sagun, L. (2021). Convit: Improving vision transformers with soft convolutional inductive biases. *arXiv preprint arXiv:2103.10697*.

Davins-Valldaura, J., Moussaoui, S., Pita-Gil, G., and Plestan, F. (2017). Parego extensions for multi-objective optimization of expensive evaluation functions. *Journal of Global Optimization*, 67(1):79–96.

Deng, J., Guo, J., Xue, N., and Zafeiriou, S. (2019). Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699.

Derringer, W. (2019). A surveillance net blankets china?s cities, giving police vast powers. *The New York Times*.

Diana, E., Gill, W., Kearns, M., Kenthapadi, K., and Roth, A. (2020). Convergent algorithms for (relaxed) minimax fairness. *arXiv preprint arXiv:2011.03108*.

Dieterich, W., Mendoza, C., and Brennan, T. (2016). Compas risk scales: Demonstrating accuracy equity and predictive parity. *Northpointe Inc*, 7(4):1–36.

Ding, F., Hardt, M., Miller, J., and Schmidt, L. (2021). Retiring adult: New datasets for fair machine learning. *Advances in neural information processing systems*, 34:6478–6490.

Donini, M., Oneto, L., Ben-David, S., Shawe-Taylor, J., and Pontil, M. (2018). Empirical risk minimization under fairness constraints. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 2796?2806.

Dooley, S. and Dickerson, J. P. (2020). The affiliate matching problem: On labor markets where firms are also interested in the placement of previous workers. *arXiv preprint arXiv:2009.11867*.

Dooley, S., Downing, R., Wei, G., Shankar, N., Thymes, B., Thorkelsdottir, G., Kurtz-Miott, T., Mattson, R., Obiwumi, O., Cherepanova, V., et al. (2021a). Comparing human and machine bias in face recognition. *arXiv preprint arXiv:2110.08396*.

Dooley, S., Goldstein, T., and Dickerson, J. P. (2021b). Robustness disparities in commercial face detection. *arXiv preprint arXiv:2108.12508*.

Dooley, S., Rosenberg, M., Sloate, E., Shin, S., and Mazurek, M. (2020). Libraries' approaches to the security of public computers. In *Proceedings of the Sixth Workshop on Inclusive Privacy and Security (WIPS 2020): In Association with the Seventeenth Symposium on Usable Privacy and Security (SOUPS 2020)*.

Dooley, S., Turjeman, D., Dickerson, J. P., and Redmiles, E. M. (2022a). Field evidence of the effects of privacy, data transparency, and pro-social appeals on covid-19 app attractiveness. In *CHI Conference on Human Factors in Computing Systems*, pages 1–21.

Dooley, S., Wei, G. Z., Goldstein, T., and Dickerson, J. (2022b). Robustness disparities in face detection. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35, pages 38245–38259. Curran Associates, Inc.

Dooley, S., Wei, G. Z., Goldstein, T., and Dickerson, J. P. (2022c). Are commercial face detection models as biased as academic models? *arXiv preprint arXiv:2201.10047*.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale.

Drozdowski, P., Rathgeb, C., Dantcheva, A., Damer, N., and Busch, C. (2020). Demographic bias in biometrics: A survey on an emerging challenge. *IEEE Transactions on Technology and Society*, 1(2):89–103.

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., and Zemel, R. (2012). Fairness through awareness. In *Innovations in Theoretical Computer Science Conference (ITCS)*.

Dwork, C. and Ilvento, C. (2018). Fairness under composition. In *Innovations in Theoretical Computer Science Conference (ITCS)*.

Edwards, H. and Storkey, A. J. (2016). Censoring representations with an adversary. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.

Eidinger, E., Enbar, R., and Hassner, T. (2014). Age and gender estimation of unfiltered faces. *IEEE Transactions on Information Forensics and Security*, 9(12):2170–2179.

El Khiyari, H. and Wechsler, H. (2016). Face verification subject to varying (age, ethnicity, and gender) demographics using deep learning. *Journal of Biometrics and Biostatistics*, 7(323):11.

Elsken, T., Metzen, J. H., and Hutter, F. (2019). Neural architecture search: A survey. *The Journal of Machine Learning Research*, 20(1):1997–2017.

Falkner, S., Klein, A., and Hutter, F. (2018). Bohb: Robust and efficient hyperparameter optimization at scale. In *International Conference on Machine Learning*, pages 1437–1446. PMLR.

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., and Venkatasubramanian, S. (2015a). Certifying and removing disparate impact. In *International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 259–268.

Feldman, M., Friedler, S. A., Moeller, J., Scheidegger, C., and Venkatasubramanian, S. (2015b). Certifying and removing disparate impact. In *Knowledge Discovery and Data Mining*, pages 259–268.

Feurer, M. and Hutter, F. (2019). Hyperparameter optimization. In *Automated machine learning*, pages 3–33. Springer, Cham.

Fitzpatrick, T. B. (1988). The validity and practicality of sun-reactive skin types i through vi. *Archives of dermatology*, 124(6):869–871.

Fu, H. and Liu, P. (2019). A multi-objective optimization model based on non-dominated sorting genetic algorithm. *International Journal of Simulation Modelling*, 18(3):510–520.

Galhotra, S., Brun, Y., and Meliou, A. (2017). Fairness testing: Testing software for discrimination. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, ESEC/FSE 2017, page 498?510, New York, NY, USA.

Garvie, C. (2016). *The perpetual line-up: Unregulated police face recognition in America*. Georgetown Law, Center on Privacy & Technology.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation.

Goel, N., Yaghini, M., and Faltings, B. (2018). Non-discriminatory machine learning through convex fairness criteria. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).

Goetzen, A., Dooley, S., and Redmiles, E. M. (2022). Ctrl-shift: How privacy sentiment changed from 2019 to 2021. *Proceedings on Privacy Enhancing Technologies*, 4:457–485.

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2015). Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*.

Google (2021a). How google uses pattern recognition to make sense of images. https://policies.google.com/technologies/pattern-recognition?hl=en-US. Accessed: 2021-06-07.

Google (2021b). How google uses pattern recognition to make sense of images.

Grgić-Hlača, N., Zafar, M. B., Gummadi, K. P., and Weller, A. (2018). Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning. In *AAAI Conference on Artificial Intelligence (AAAI)*.

Grother, P., Ngan, M., and Hanaoka, K. (2019). *Face Recognition Vendor Test (FVRT): Part 3, Demographic Effects*. National Institute of Standards and Technology.

Guo, Z., Zhang, X., Mu, H., Heng, W., Liu, Z., Wei, Y., and Sun, J. (2020). Single path one-shot neural architecture search with uniform sampling. In *European conference on computer vision*, pages 544–560. Springer.

Gutman, D. (2021). King County Council bans use of facial recognition technology by Sheriff's Office, other agencies. *The Seattle Times*.

Hamidi, F., Scheuerman, M. K., and Branham, S. M. (2018). Gender recognition or gender reductionism? the social implications of embedded gender recognition systems. In *Proceedings of the 2018 chi conference on human factors in computing systems*, pages 1–13.

Han, D., Kim, J., and Kim, J. (2017). Deep pyramidal residual networks. *IEEE CVPR*.

Han, D., Yun, S., Heo, B., and Yoo, Y. (2020a). Rexnet: Diminishing representational bottleneck on convolutional neural network. *arXiv preprint arXiv:2007.00992*, 6.

Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., and Xu, C. (2020b). Ghostnet: More features from cheap operations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1580–1589.

Han, K., Xiao, A., Wu, E., Guo, J., Xu, C., and Wang, Y. (2021). Transformer in transformer. *Advances in Neural Information Processing Systems*, 34:15908–15919.

Hanna, A., Denton, E., Smart, A., and Smith-Loud, J. (2020). Towards a critical race methodology in algorithmic fairness. In *ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pages 501–512.

Hardt, M., Price, E., and Srebro, N. (2016). Equality of opportunity in supervised learning. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Hartzog, W. (2020). The secretive company that might end privacy as we know it. *The New York Times*.

Hashimoto, T. B., Srivastava, M., Namkoong, H., and Liang, P. (2018). Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning (ICML)*.

Hazirbas, C., Bitton, J., Dolhansky, B., Pan, J., Gordo, A., and Ferrer, C. C. (2021). Towards measuring fairness in ai: the casual conversations dataset. *arXiv preprint arXiv:2104.02821*.

He, K., Zhang, X., Ren, S., and Sun, J. (2016a). Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

He, K., Zhang, X., Ren, S., and Sun, J. (2016b). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Heidari, H. and Krause, A. (2018). Preventing disparate treatment in sequential decision making. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*.

Heidari, H., Nanda, V., and Gummadi, K. P. (2019). On the long-term impact of algorithmic decision policies: Effort unfairness and feature segregation through social learning. In *International Conference on Machine Learning (ICML)*.

Hendrycks, D. and Dietterich, T. (2019). Benchmarking neural network robustness to common corruptions and perturbations. In *Proceedings of the International Conference on Learning Representations*.

Hill, K. (2020a). Another arrest, and jail time, due to a bad facial recognition match. *The New York Times*, 29.

Hill, K. (2020b). The secretive company that might end privacy as we know it. In *Ethics of Data and Analytics*, pages 170–177. Auerbach Publications.

Holstein, K., Wortman Vaughan, J., Daumé, H., Dudik, M., and Wallach, H. (2019). Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1?16.

Hosseini, H., Xiao, B., and Poovendran, R. (2017). Google's cloud vision API is not robust to noise. In *2017 16th IEEE international conference on machine learning and applications (ICMLA)*, pages 101–105. IEEE.

Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., Pang, R., Vasudevan, V., et al. (2019). Searching for mobilenetv3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1314–1324.

Hu, J., Shen, L., and Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141.

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Computer Vision and Pattern Recognition (CVPR)*.

Huang, G. B., Mattar, M., Berg, T., and Learned-Miller, E. (2008). Labeled faces in the wild: A database forstudying face recognition in unconstrained environments. In *Workshop on faces in'Real-Life'Images: detection, alignment, and recognition*.

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5mb model size. *CoRR*, abs/1602.07360.

Irvin, J., Rajpurkar, P., Ko, M., Yu, Y., Ciurea-Ilcus, S., Chute, C., Marklund, H., Haghgoo, B., Ball, R., Shpanskaya, K., Seekins, J., Mong, D. A., Halabi, S. S., Sandberg, J. K., Jones, R., Larson, D. B., Langlotz, C. P., Patel, B. N., Lungren, M. P., and Ng, A. Y. (2019). Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33:590–597.

Jain, G. and Parsheera, S. (2021). 1.4 billion missing pieces? auditing the accuracy of facial processing tools on indian faces. *First Workshop on Ethical Considerations in Creative applications of Computer Vision*.

Jaiswal, S., Duggirala, K., Dash, A., and Mukherjee, A. (2022). Two-face: Adversarial audit of commercial face recognition systems. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 16, pages 381–392.

Joo, J. and Kärkkäinen, K. (2020). Gender slopes: Counterfactual fairness for computer vision models by attribute manipulation. *arXiv preprint arXiv:2005.10430*.

Käding, C., Rodner, E., Freytag, A., and Denzler, J. (2016). Fine-tuning deep neural networks in continuous learning scenarios. In *Asian Conference on Computer Vision*, pages 588–605. Springer.

Kantayya, S. (2020). Coded bias. Feature-length documentary.

Keyes, O. (2018). The misgendering machines: Trans/hci implications of automatic gender recognition. *Proceedings of the ACM on human-computer interaction*, 2(CSCW):1–22.

Khandani, A. E., Kim, A. J., and Lo, A. W. (2010). Consumer credit-risk models via machine-learning algorithms. *Journal of Banking & Finance*, 34(11):2767–2787.

Khani, F. and Liang, P. (2019). Noise induces loss discrepancy across groups for linear regression.

Khurana, G. S., Dooley, S., Naidu, S. V., and White, C. (2023). Forecastpfn: Universal forecasting for healthcare. In *ICLR 2023 Workshop on Time Series Representation Learning for Health*.

Kilbertus, N., Carulla, M. R., Parascandolo, G., Hardt, M., Janzing, D., and Schölkopf, B. (2017). Avoiding discrimination through causal reasoning. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 656–666.

Klare, B. F., Burge, M. J., Klontz, J. C., Bruegge, R. W. V., and Jain, A. K. (2012). Face recognition performance: Role of demographic information. *IEEE Transactions on Information Forensics and Security*, 7(6):1789–1801.

Knittel, M., Dooley, S., and Dickerson, J. P. (2022). The dichotomous affiliate stable matching problem: Approval-based matching with applicant-employer relations. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*.

Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. *Master's thesis, University of Toronto*.

Krizhevsky, A. (2014). One weird trick for parallelizing convolutional neural networks. *CoRR*, abs/1404.5997.

Kuo, K., Ostuni, A., Horishny, E., Curry, M. J., Dooley, S., Chiang, P.-y., Goldstein, T., and Dickerson, J. P. (2020). Proportionnet: Balancing fairness and revenue for auction design with deep learning. *arXiv preprint arXiv:2010.06398*.

Kusner, M. J., Loftus, J., Russell, C., and Silva, R. (2017). Counterfactual fairness. In *Advances in Neural Information Processing Systems 30*, pages 4066–4076.

Lahoti, P., Beutel, A., Chen, J., Lee, K., Prost, F., Thain, N., Wang, X., and Chi, E. H. (2020). Fairness without demographics through adversarially reweighted learning. *arXiv preprint arXiv:2006.13114*.

Laumanns, M. and Ocenasek, J. (2002). Bayesian optimization algorithms for multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature*, pages 298–307. Springer.

Learned-Miller, E., Ordóñez, V., Morgenstern, J., and Buolamwini, J. (2020). Facial recognition technologies in the wild.

Leben, D. (2020). Normative principles for evaluating fairness in machine learning. In *Conference on Artificial Intelligence, Ethics, and Society (AIES)*, pages 86–92.

Lee, Y. and Park, J. (2020). Centermask: Real-time anchor-free instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13906–13915.

Leslie, D. (2020). Understanding bias in facial recognition technologies. *arXiv preprint arXiv:2010.07023*.

Lewis, S. (2019). The racial bias built into photography. *The New York Times*, 25.

Li, H., Lin, Z., Shen, X., Brandt, J., and Hua, G. (2015). A convolutional neural network cascade for face detection. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5325–5334.

Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., and Talwalkar, A. (2017). Hyperband: A novel bandit-based approach to hyperparameter optimization. *The Journal of Machine Learning Research*, 18(1):6765–6816.

Lin, T.-Y., Dollar, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Lin, X., Kim, S., and Joo, J. (2022). Fairgrape: Fairness-aware gradient pruning method for face attribute classification. *arXiv preprint arXiv:2207.10888*.

Lindauer, M., Eggensperger, K., Feurer, M., Biedenkapp, A., Deng, D., Benjamins, C., Ruhkopf, T., Sass, R., and Hutter, F. (2022). Smac3: A versatile bayesian optimization package for hyperparameter optimization. *Journal of Machine Learning Research*, 23(54):1–9.

Liu, H., Simonyan, K., and Yang, Y. (2018a). Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*.

Liu, L. T., Dean, S., Rolf, E., Simchowitz, M., and Hardt, M. (2018b). Delayed impact of fair machine learning. In *International Conference on Machine Learning (ICML)*.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. *Lecture Notes in Computer Science*, page 21–37.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738.

Lohr, S. (2018). Facial recognition is accurate, if you?re a white guy. *New York Times*, 9.

Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Luo, A. F., Warford, N., Dooley, S., Greenstadt, R., Mazurek, M. L., and McDonald, N. (2023). How library it staff navigate privacy and security challenges and responsibilities. In *USENIX Security*.

Madras, D., Creager, E., Pitassi, T., and Zemel, R. S. (2018). Learning adversarially fair and transferable representations. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 3381–3390. PMLR.

Majumdar, P., Mittal, S., Singh, R., and Vatsa, M. (2021). Unravelling the effect of image distortions for biased prediction of pre-trained face recognition models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3786–3795.

Mao-Guo, G., Li-Cheng, J., Dong-Dong, Y., and Wen-Ping, M. (2009). Evolutionary multi-objective optimization algorithms. *Journal of Software*, 20(2).

Marson, J. and Forrest, B. (2021). Armed low-cost drones, made by turkey, reshape battlefields and geopolitics. *The Wall Street Journal*.

Martinez, N., Bertran, M., and Sapiro, G. (2020). Minimax pareto fairness: A multi objective perspective. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pages 6755–6764.

Mehrabi, N., Naveed, M., Morstatter, F., and Galstyan, A. (2020). Exacerbating algorithmic bias through fairness attacks.

Mehta, D., Sotnychenko, O., Mueller, F., Xu, W., Elgharib, M., Fua, P., Seidel, H.-P., Rhodin, H., Pons-Moll, G., and Theobalt, C. (2019). Xnect: Real-time multi-person 3d human pose estimation with a single rgb camera. *arXiv preprint arXiv:1907.00837*.

Meng, Q., Zhao, S., Huang, Z., and Zhou, F. (2021). Magface: A universal representation for face recognition and quality assessment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14225–14234.

Minaee, S., Luo, P., Lin, Z., and Bowyer, K. (2021). Going deeper into face detection: A survey.

Moosavi-Dezfooli, S.-M., Fawzi, A., and Frossard, P. (2016). Deepfool: a simple and accurate method to fool deep neural networks. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2574–2582.

Morales, A., Fierrez, J., Vera-Rodriguez, R., and Tolosana, R. (2020). Sensitivenets: Learning agnostic representations with application to face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Moschoglou, S., Papaioannou, A., Sagonas, C., Deng, J., Kotsia, I., and Zafeiriou, S. (2017). Agedb: the first manually collected, in-the-wild age database. In *proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 51–59.

Mukerjee, A., Biswas, R., Deb, K., and Mathur, A. P. (2002). Multi–objective evolutionary algorithms for the risk–return trade–off in bank loan management. *International Transactions in operational research*.

Nanda, V., Dooley, S., Singla, S., Feizi, S., and Dickerson, J. P. (2021). Fairness through robustness: Investigating robustness disparity in deep learning. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 466–477.

Narayanan, A. (2018). Translation tutorial: 21 fairness definitions and their politics. In *Proc. Conf. Fairness Accountability Transp., New York, USA*.

Ngai, E. W., Hu, Y., Wong, Y. H., Chen, Y., and Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision support systems*, 50(3):559–569.

of Washington 66th Legislature 2020 Regular Session, S. (2020). Senate bill 6281.

O'Toole, A. J., Phillips, P. J., An, X., and Dunlop, J. (2012). Demographic effects on estimates of automatic face recognition performance. *Image and Vision Computing*, 30(3):169–176.

Ouyang, W., Wang, X., Zhang, C., and Yang, X. (2016). Factors in finetuning deep model for object detection with long-tail distribution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 864–873.

Padala, M. and Gujar, S. (2020). Fnnc: Achieving fairness through neural networks. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 2277–2283. International Joint Conferences on Artificial Intelligence Organization.

Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z. B., and Swami, A. (2016). The limitations of deep learning in adversarial settings. In *IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 372–387. IEEE.

Paria, B., Kandasamy, K., and Póczos, B. (2020). A flexible framework for multi-objective bayesian optimization using random scalarizations. In *Uncertainty in Artificial Intelligence*, pages 766–776. PMLR.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, pages 8026–8037.

Patrick J. Grother, George W. Quinn, P. J. P. (2010). Report on the evaluation of 2d still-image face recognition algorithms. *NIST Interagency/Internal Report (NISTIR)*.

Peri, N., Curry, M., Dooley, S., and Dickerson, J. (2021). Preferencenet: Encoding human preferences in auction design with deep learning. *Advances in Neural Information Processing Systems*, 34:17532–17542.

Perrone, V., Donini, M., Zafar, M. B., Schmucker, R., Kenthapadi, K., and Archambeau, C. (2021). Fair bayesian optimization. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 854–863.

Pham, H., Guan, M., Zoph, B., Le, Q., and Dean, J. (2018). Efficient neural architecture search via parameters sharing. In *International conference on machine learning*, pages 4095–4104. PMLR.

Phillips, P. J., Beveridge, J. R., Draper, B. A., Givens, G., O'Toole, A. J., Bolme, D. S., Dunlop, J., Lui, Y. M., Sahibzada, H., and Weimer, S. (2011). An introduction to the good, the bad, & the ugly face recognition challenge problem. In *2011 IEEE International Conference on Automatic Face & Gesture Recognition (FG)*, pages 346–353. IEEE.

Phillips, P. J., Scruggs, W. T., O?Toole, A. J., Flynn, P. J., Bowyer, K. W., Schott, C. L., and Sharpe, M. (2007). Frvt 2006 and ice 2006 large-scale results. *National Institute of Standards and Technology, NISTIR*, 7408(1):1.

Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., and Weinberger, K. Q. (2017). On fairness and calibration. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 5680–5689. Curran Associates, Inc.

Quadrianto, N., Sharmanska, V., and Thomas, O. (2019). Discovering fair representations in the data domain. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 8227–8236. Computer Vision Foundation / IEEE.

Raji, I. D. and Buolamwini, J. (2019). Actionable auditing: Investigating the impact of publicly naming biased performance results of commercial ai products. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 429–435.

Raji, I. D. and Fried, G. (2021). About face: A survey of facial recognition evaluation. *arXiv preprint arXiv:2102.00813*.

Raji, I. D., Gebru, T., Mitchell, M., Buolamwini, J., Lee, J., and Denton, E. (2020). Saving face: Investigating the ethical concerns of facial recognition auditing. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 145–151.

Rawson, M., Dooley, S., Bharadwaj, M., and Choudhary, R. (2022). Topological data analysis for word sense disambiguation. *arXiv preprint arXiv:2203.00565*.

Ribeiro, F. N., Saha, K., Babaei, M., Henrique, L., Messias, J., Benevenuto, F., Goga, O., Gummadi, K. P., and Redmiles, E. M. (2019). On microtargeting socially divisive ads: A case study of russia-linked ad campaigns on facebook. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, page 140?149, New York, NY, USA.

Ruane, K. (2021). Biden must halt face recognition technology to advance racial equity. *ACLU*.

Ryu, H. J., Adam, H., and Mitchell, M. (2018). Inclusivefacenet: Improving face attribute detection with race and gender diversity. *arXiv preprint arXiv:1712.00193*.

Saha, D., Schumann, C., McElfresh, D. C., Dickerson, J. P., Mazurek, M. L., and Tschantz, M. C. (2020). Measuring non-expert comprehension of machine learning fairness metrics. In *International Conference on Machine Learning (ICML)*.

Salman, H., Li, J., Razenshteyn, I., Zhang, P., Zhang, H., Bubeck, S., and Yang, G. (2019). Provably robust deep learning via adversarially trained smoothed classifiers. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 11292–11303.

Savani, Y., White, C., and Govindarajulu, N. S. (2020). Intra-processing methods for debiasing neural networks. In *Proceedings of Advances in Neural Information Processing Systems*.

Schmucker, R., Donini, M., Perrone, V., Zafar, M. B., and Archambeau, C. (2020). Multi-objective multi-fidelity hyperparameter optimization with application to fairness. In *NeurIPS Workshop on Meta-Learning*, volume 2.

Schmucker, R., Donini, M., Zafar, M. B., Salinas, D., and Archambeau, C. (2021). Multi-objective asynchronous successive halving. *arXiv preprint arXiv:2106.12639*.

Schumann, C., Counts, S. N., Foster, J. S., and Dickerson, J. P. (2019). The diverse cohort selection problem. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, page 601?609.

Schumann, C., Foster, J. S., Mattei, N., and Dickerson, J. P. (2020). We need fairness and explainability in algorithmic hiring. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, page 1716?1720.

Schumann, C., Pantofaru, C. R., Ricco, S., Prabhu, U., and Ferrari, V. (2021). A step toward more inclusive people annotations for fairness. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*.

Selbst, A. D., Boyd, D., Friedler, S. A., Venkatasubramanian, S., and Vertesi, J. (2019). Fairness and abstraction in sociotechnical systems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, page 59–68, New York, NY, USA. Association for Computing Machinery.

Sengupta, S., Chen, J.-C., Castillo, C., Patel, V. M., Chellappa, R., and Jacobs, D. W. (2016). Frontal to profile face verification in the wild. In *2016 IEEE winter conference on applications of computer vision (WACV)*, pages 1–9. IEEE.

Shan, S., Wenger, E., Zhang, J., Li, H., Zheng, H., and Zhao, B. Y. (2020). Fawkes: Protecting privacy against unauthorized deep learning models. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pages 1589–1604.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*.

Singer, N. (2018). Microsoft urges congress to regulate use of facial recognition. *The New York Times*.

Singh, A. and Joachims, T. (2018). Fairness of exposure in rankings. In *International Conference on Knowledge Discovery and Data Mining (KDD).*

Singh, R., Agarwal, A., Singh, M., Nagpal, S., and Vatsa, M. (2020). On the robustness of face recognition algorithms against attacks and bias. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13583–13589.

Singh, R., Majumdar, P., Mittal, S., and Vatsa, M. (2022). Anatomizing bias in facial analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 12351–12358.

Singla, S. and Feizi, S. (2020). Second-order provable defenses against adversarial attacks. In *International Conference on Machine Learning (ICML).*

Solans, D., Biggio, B., and Castillo, C. (2020). Poisoning attacks on algorithmic fairness.

Speicher, T., Ali, M., Venkatadri, G., Ribeiro, F. N., Arvanitakis, G., Benevenuto, F., Gummadi, K. P., Loiseau, P., and Mislove, A. (2018). Potential for discrimination in online targeted advertising. In *ACM Conference on Fairness, Accountability, and Transparency (FAccT).*

Sun, K., Xiao, B., Liu, D., and Wang, J. (2019). Deep high-resolution representation learning for human pose estimation. In *CVPR*.

Suykens, J. A. K. and Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293?300.

Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first AAAI conference on artificial intelligence*.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2014). Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR).*

Tan, M. and Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR.

Too, E. C., Yujian, L., Njuki, S., and Yingchun, L. (2019). A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*, 161:272–279.

Tramer, F., Carlini, N., Brendel, W., and Madry, A. (2020). On adaptive attacks to adversarial example defenses.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Verma, S. and Rubin, J. (2018). Fairness definitions explained. In *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)*, pages 1–7. IEEE.

Wadsworth, C., Vera, F., and Piech, C. (2018). Achieving fairness through adversarial learning: an application to recidivism prediction. *CoRR*, abs/1807.00199.

Wang, C.-Y., Liao, H.-Y. M., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., and Yeh, I.-H. (2020a). Cspnet: A new backbone that can enhance learning capability of cnn. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 390–391.

Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., and Liu, W. (2018). Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5265–5274.

Wang, M. and Deng, W. (2018). Deep face recognition: A survey. *arXiv preprint arXiv:1804.06655*.

Wang, M. and Deng, W. (2020). Mitigating bias in face recognition using skewness-aware reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9322–9331.

Wang, M., Deng, W., Hu, J., Tao, X., and Huang, Y. (2019a). Racial faces in the wild: Reducing racial bias by information maximization adaptation network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 692–702.

Wang, Q., Zhang, P., Xiong, H., and Zhao, J. (2021). Face. evolve: A high-performance face recognition library. *arXiv preprint arXiv:2107.08621*.

Wang, T., Zhao, J., Yatskar, M., Chang, K.-W., and Ordonez, V. (2019b). Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5310–5319.

Wang, X. (2021). Teacher guided neural architecture search for face recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2817–2825.

Wang, Z., Qinami, K., Karakozis, Y., Genova, K., Nair, P., Hata, K., and Russakovsky, O. (2020b). Towards fairness in visual recognition: Effective strategies for bias mitigation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8916–8925.

Weise, K. and Singer, N. (2020). Amazon pauses police use of its facial recognition software. *The New York Times*.

White, C., Neiswanger, W., and Savani, Y. (2021). Bananas: Bayesian optimization with neural architectures for neural architecture search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10293–10301.

Wightman, R. (2019). Pytorch image models. https://github.com/rwightman/pytorch-image-models.

Wilber, M. J., Shmatikov, V., and Belongie, S. (2016). Can we still avoid automatic face detection? In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9. IEEE.

Wu, W., Protopapas, P., Yang, Z., and Michalatos, P. (2020). Gender classification and bias mitigation in facial images. In *12th acm conference on web science*, pages 106–114.

Xie, S., Girshick, R. B., Dollár, P., Tu, Z., and He, K. (2016). Aggregated residual transformations for deep neural networks. *CoRR*, abs/1611.05431.

Xiong, Y., Zhu, K., Lin, D., and Tang, X. (2015). Recognize complex events from static images by fusing deep channels. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*. IEEE.

Xu, W., Xu, Y., Chang, T., and Tu, Z. (2021). Co-scale conv-attentional image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9981–9990.

Xu, Y., Xie, L., Zhang, X., Chen, X., Qi, G.-J., Tian, Q., and Xiong, H. (2019). Pc-darts: Partial channel connections for memory-efficient architecture search. *arXiv preprint arXiv:1907.05737*.

Yang, S., Luo, P., Loy, C. C., and Tang, X. (2016). Wider face: A face detection benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yu, F., Wang, D., Shelhamer, E., and Darrell, T. (2018). Deep layer aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2403–2412.

Zafar, M. B., Valera, I., Gomez Rodriguez, M., and Gummadi, K. P. (2017a). Fairness beyond disparate treatment & disparate impact. *Proceedings of the 26th International Conference on World Wide Web*.

Zafar, M. B., Valera, I., Gomez-Rodriguez, M., and Gummadi, K. P. (2017b). Fairness constraints: Mechanisms for fair classification. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, volume 54 of *Proceedings of Machine Learning Research*, pages 962–970. PMLR.

Zafar, M. B., Valera, I., Gomez-Rodriguez, M., and Gummadi, K. P. (2019). Fairness constraints: A flexible approach for fair classification. *Journal of Machine Learning Research*, 20(75):1–42.

Zafar, M. B., Valera, I., Rodriguez, M. G., Gummadi, K. P., and Weller, A. (2017c). From parity to preference-based notions of fairness in classification. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*.

Zela, A., Elsken, T., Saikia, T., Marrakchi, Y., Brox, T., and Hutter, F. (2019). Understanding and robustifying differentiable architecture search. *arXiv preprint arXiv:1909.09656*.

Zemel, R., Wu, Y., Swersky, K., Pitassi, T., and Dwork, C. (2013). Learning fair representations. In *International conference on machine learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 325–333, Atlanta, Georgia, USA. PMLR.

Zhang, Z., Song, Y., and Qi, H. (2017). Age progression/regression by conditional adversarial autoencoder. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5810–5818.

Zhang, Z., Zhang, H., Zhao, L., Chen, T., and Pfister, T. (2021). Aggregating nested transformers. *arXiv preprint arXiv:2105.12723*.

Zheng, T. and Deng, W. (2018). Cross-pose lfw: A database for studying cross-pose face recognition in unconstrained environments. *Beijing University of Posts and Telecommunications, Tech. Rep*, 5(7).

Zheng, T., Deng, W., and Hu, J. (2017). Cross-age lfw: A database for studying cross-age face recognition in unconstrained environments. *arXiv preprint arXiv:1708.08197*.

Zhu, N. (2019). Neural architecture search for deep face recognition. *arXiv preprint arXiv:1904.09523*.