Debiasify: Self-Distillation for Unsupervised Bias Mitigation

Nourhan Bayasi[1]* Jamil Fayyad[2]* Ghassan Hamarneh[3] Rafeef Garbi[1] Homayoun Najjaran[2]
[1]University of British Columbia [2]University of Victoria [3]Simon Fraser University
nourhanb,rafeef@ece.ubc.ca jfayyad,najjaran@uvic.ca hamarneh@sfu.ca

Abstract

Simplicity bias is a critical challenge in neural networks since it often leads to favoring simpler solutions and learning unintended decision rules captured by spurious correlations, causing models to be biased and diminishing their generalizability. While existing solutions rely on human supervision, obtaining annotations of the different bias attributes is often impractical. To tackle this, we present `Debiasify`*, a novel self-distillation approach that works without any prior information about the nature of biases. Our method leverages a new distillation loss to distill knowledge within a network; from a deep layer where complex, highly-predictive features reside, to a shallow layer where simpler yet attribute-conditioned features are found in an unsupervised manner. In this way,* `Debiasify` *learns robust, debiased representations that generalize well across various biases and datasets, enhancing worst-group performance and overall accuracy. Extensive experiments on computer vision and medical imaging benchmarks show the efficacy of our method, significantly outperforming the previous unsupervised debiasing methods (e.g., a 10.13% improvement in worst-group accuracy on Wavy Hair classification in CelebA) while achieving comparable or superior performance to supervised methods. Our code is publicly available at the following link:Debiasify.*

1. Introduction

Deep neural networks have emerged as a fundamental technology in numerous applications that profoundly impact various aspects of society, such as facial recognition [17], AI-enabled recruitment [20], and healthcare diagnostics [3, 11]. Given the significant societal implications of these algorithms, it is increasingly crucial to ensure their resilience against *simplicity bias* [4, 31, 35]; in other words, these networks' learning process should not prioritize weak predictive features over complex features that underpin the actual mechanisms of the task of interest. For instance, on the CelebA dataset [24], which is a real-world dataset where different attributes are strongly correlated, networks tend to classify hair color based on gender, frequently associating `Blond Hair` with `Female`. Such an unintended rule performs adequately across the majority of training instances but leads to unforeseen extreme errors in minority examples that lack the spurious correlation, thereby hindering the model's ability to adapt to new testing scenarios that exhibit changes in data distributions.

Effective ways for network debiasing include upweighting or upsampling of examples that lack spurious correlations [29], data augmentation [12], adversarial learning [11,37], robust learning [31], and architecture optimization [2]. Nevertheless, most of these efforts rely on explicit bias attribute labels in their debiasing recipes. This compromises their practicality, as identifying and manually labeling the types of biases, to determine which attributes involve spurious correlations without a thorough analysis of the model and dataset, present significant challenges. Only recently, the focus has been shifted towards debiasing without the bias attribute labels. This is usually achieved by identifying the minority group within each class – flagged based on indicators such as misclassification [22], high loss [28], or sensitive representations [9], and subsequently upweighting/upsampling them during training.

Despite being promising, these methods have two major drawbacks. First, they are heavily dependent on hyperparameter tuning using bias attribute information in the validation set, which might not be accessible for datasets in the real world [19]. Second, they are designed to address only a single bias attribute within a class, neglecting the potential existence of multiple bias sources within the same class; e.g., skin tone, gender, image background.

To overcome the aforementioned problems, we propose `Debiasify`, a simple yet effective unsupervised debiasing technique via feature clustering and self-distillation. Rooted in the observations that images sharing the same label for certain bias attribute(s) (other than the target attribute) tend to have similar representations in the feature space, particularly in the shallow layers of the neural network stack [10,37], we propose clustering the shallow layer

*Equal contribution.

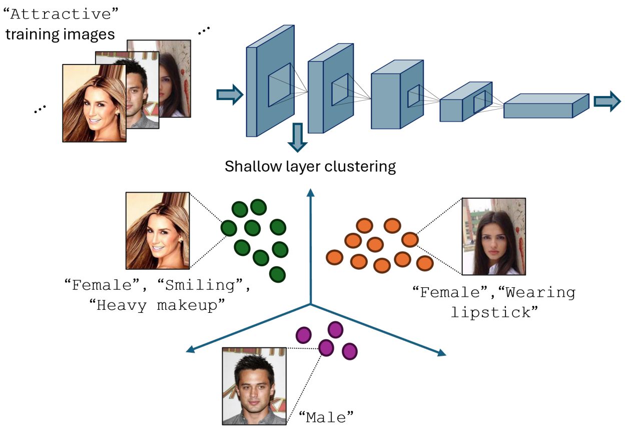


Figure 1. Debiasify leverages clustering in the feature space of a shallow layer in the network to identify attribute-conditioned groups (3 groups shown) for each class (e.g., Attractive), where images in each group are clustered based on common, non-target bias attributes (e.g., Female, Smiling, etc.)

features to identify attribute-conditioned groups for each class, as depicted in Figure 1. To exploit these groups for learning debiased representations, we introduce a novel self-distillation loss that encourages their distributions to converge while simultaneously aligning them with the distribution of their highly-predictive class features in the deep layer. In summary, our contributions are:

- We introduce Debiasify, a new method for unsupervised bias mitigation through a self, deep-to-shallow, distillation technique.
- We propose a hybrid loss that maintains high classification performance while effectively debiasing representations by minimizing the distance between class-specific, bias attribute-conditioned groups in the shallow layers and their corresponding class attribute-agnostic distributions in the deep layers.
- We conduct experiments and ablation studies on CelebA, Waterbirds, and Fitzpatrick, benchmarking against bias-unsupervised methods, including previous SOTA: CFix [7], and the uper-bound supervised method: GDRO [32]. Our results highlight Debiasify's superior performance, especially in worst-group accuracy.

2. Related Work

2.1. Simplicity Bias in Neural Networks

Neural networks have been found to be prone to simplicity bias [5, 35]. That is, they tend to learn the simplest features to solve a task, even in the presence of other, more robust but more complex features. This bias towards simpler features can lead to models lacking robustness against shifts that do not adhere to the simplistic characteristics captured by the learned features. Extensive efforts have been made to address the simplicity bias problem, categorized broadly into three approaches based on the stage of intervention during the modeling process. Pre-processing techniques aim to modify the training data in order to reduce the correlations between bias and target attributes [6, 26]; in-processing techniques modify the learning algorithms to eliminate bias during the model training process [8, 11]; and post-processing techniques treat the learned model as a black-box and try to mitigate bias by leveraging the predictions [30, 41]. Nevertheless, most of these techniques have limitations in real-world scenarios since they rely on access to bias attribute annotations in the training or validation sets for effective bias mitigation.

2.2. Bias Mitigation without Supervision

Recently, efforts have been directed towards mitigating bias in the absence of explicit bias attributes, primarily through in-processing techniques [1, 3, 27, 37, 39, 42]. One such technique involves identifying minority samples as those misclassified by an initial network and then reweighting them. Nam et al. [28] achieve this by training an additional biased model, where images that are not easily trained by the biased model are considered minority. Liu et al. [22] define minority samples as those misclassified by a model trained using empirical risk minimization and prioritize them during the training of a debiased model. Another technique synthesizes images having similar characteristics to the minority group and employs them to train a debiased model. Kim et al. [18] synthesize images without bias attributes by leveraging an image-to-image translation model. Lee et al. [21] and Hwang et al. [16] augment minority samples in the feature space by employing disentangled representations and mixup, respectively.

Our work aligns closely with a third technique, which involves acquiring bias pseudo-labels through the unsupervised learning technique of clustering in the feature space of the network. Examples in this category include BPA [34], which proposes a cluster-wise reweighting scheme, leveraging pseudo-attribute information from feature clustering results; CFix [7], which uses cluster error (i.e., the discrepancy in correctly classifying examples within clusters) to identify examples potentially influenced by network inductive bias, subsequently upweighting them to enhance worst-group performance; and George [36], which approximates bias attributes with cluster assignment and weights the objective function to maximize the worst-group accuracy. However, these clustering-based methods have a key limitation: they rely solely on reweighting of images within their respective clusters, which can be problematic given

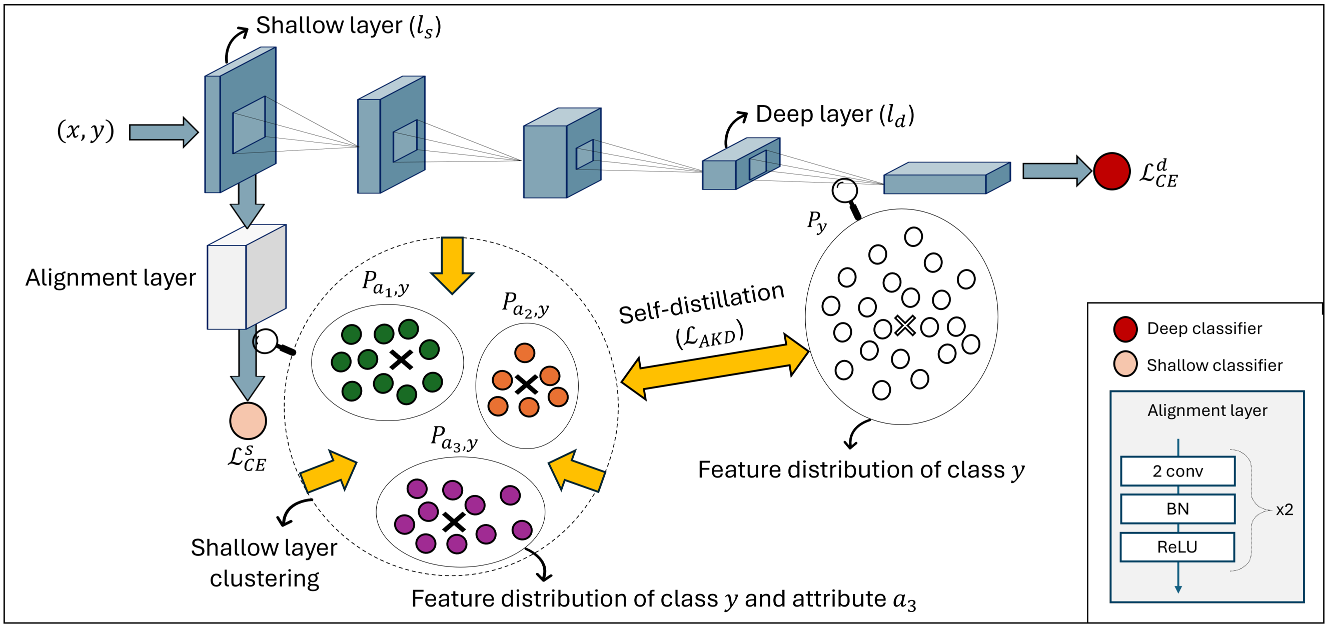


Figure 2. Debiasify identifies attribute-conditioned groups $P_{a_k,y}$ (represented by ● ● ● for all a_k attributes, k = 1, 2, 3) found through clustering in feature space of a shallow network layer. The goal is to bring their distributions closer to each other while aligning them with their class distribution P_y (represented by ○) in the deep layer using a novel self-distillation loss $\mathcal{L}_{AKD}$ (yellow arrows).

that formation of clusters (their size and shape) is sensitive to outliers and noisy images that are commonly encountered in real-world datasets. In contrast, our Debiasify takes a fundamentally different approach by naturally examining cluster distributions and promoting their alignment through a novel self-distillation loss, without any reweighting.

3. Methodology

Figure 2 presents an overview of our proposed self-distillation method for unsupervised bias mitigation, termed Debiasify. Our self-distillation loss directs a shallow network layer to learn more predictive features instead of simpler ones that might be correlated with unwanted characteristics to improve performance. Assuming bias information is unknown, Debiasify first performs clustering in the feature space of a shallow layer to identify class-wise attribute-conditioned groups, which are then guided to get closer to each other while simultaneously mirroring their class, attribute-agnostic counterparts in the deep layer. Observations and details are given next.

3.1. Observations

Objective. Let $\mathbf{x}$ be an image associated with a set of possible bias attributes $a \in \mathcal{A}$. The primary objective of our Debiasify is to predict a target attribute y by estimating the ground truth relationship $p(y \mid \mathbf{x})$, while mitigating any undesired correlations with other bias attributes; i.e., ensuring that $p(y \mid \mathbf{x}) = p(y \mid \mathbf{x}, a), \forall\, a \in \mathcal{A}$. During the training phase of Debiasify, information regarding bias set $\mathcal{A}$ is neither available nor provided.

Preliminary Study. Previous works [7, 34] have used the CelebA dataset to analyze the feature semantics over the target and bias attributes. The studies revealed that images from certain groups, defined by a combination of target and bias attribute values, are clustered in the feature space, even without using the bias information during training. Expanding on their analysis, we conduct the following experiments to investigate the layer where the clustering could be more pronounced. We begin by training a ResNet18 model for 50 epochs to classify a target attribute (e.g., Blond Hair). Next, we assess feature decodability at different network depths to evaluate how well the bias attribute Male can be decoded, keeping the network parameters frozen. A decoder, consisting of a single linear layer and softmax, is trained using activations from a specific layer and an unbiased validation set labeled with the bias attribute. Figure 3 – (a, blue bars) shows the decodability of Male across layers. We observe that bias decodability generally decreases with network depth, indicating that shallower layers are more effective at detecting bias. In more complex scenarios, where the bias involves combinations of attributes, a similar trend is observed (Figure 3 – (b, blue bars)). These findings align with previous studies that identified bias detection in shallow layers [10, 37, 38] and were theoretically validated [15].

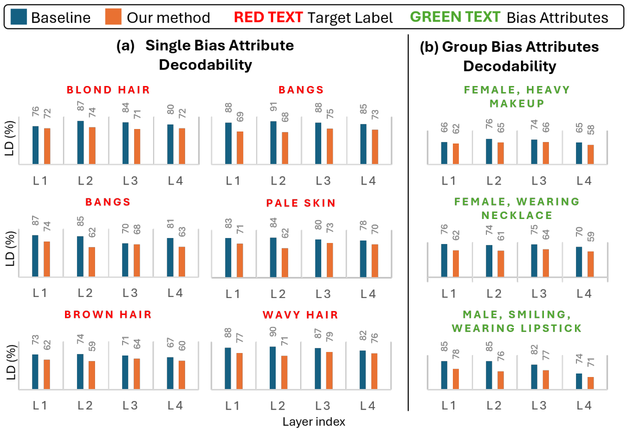


Figure 3. Results of Linear Decodability (LD): Panel (a) compares LD of the `Male` bias attribute from a frozen baseline network (blue) and our method (orange), both pretrained on different target attributes. Panel (b) shows LD of multiple bias attributes from networks pretrained on `Blond Hair`, comparing the baseline (blue) and our method (orange).

3.2. Formulation

Preliminaries. We aim to train a network f to enhance the classification performance and generalization by learning debiased representations. We denote the shallow and deep layers of f as l_s and l_d, respectively. To align the feature distributions across l_s and l_d, a simple auxiliary branch is introduced after l_s, comprising an alignment layer and a classifier, as illustrated in Figure 2. The alignment layer ensures that the feature dimensionality in l_s matches that of l_d. Additionally, we denote the predicted labels of sample $\mathbf{x}$ by the shallow and deep classifiers as c_s and c_d, respectively.

Generating Attribute-conditioned Groups. We begin by training f for a few epochs using an averaged cross entropy loss $\mathcal{L}_{ACE}$ to predict the ground truth y for each sample $\mathbf{x}$, as follows;

$$\mathcal{L}_{ACE} = \frac{1}{2} \sum_{y} \left[\mathcal{L}_{CE}^{s} \left(c_s, y \right) + \mathcal{L}_{CE}^{d} \left(c_d, y \right) \right] . \qquad (1)$$

Once the network is trained and potentially learned the bias, we cluster the feature embeddings from l_s to generate $\mathcal{K}$ attribute-conditioned groups for each class, given our empirical observations in Sec 3.1 and previous findings confirming that clustering effectively groups images sharing common non-target attributes. In our implementations, we opt to use K-means [25] with adaptive $\mathcal{K}$ value to mitigate the risk of failing to capture smaller clusters if $\mathcal{K}$ is small or the emergence of unwanted clusters if $\mathcal{K}$ is large. We determine the value of $\mathcal{K}$ as the smallest integer such that the mean within-cluster variance of the features is lower than a pre-defined upper bound γ. As a result, we obtain a set of clusters that comprehensively cover each class, where images within each cluster share common attributes other than the target attribute.

Learning Debiased Representations. Next, we aim to learn debiased representations by minimizing the discrepancy between the class-wise feature distribution P_y extracted from l_d, and the distribution of each attribute-conditioned cluster $P_{a_k,y}$ derived from l_s and associated with the same class. Note that k ranges from 1 to $\mathcal{K}$. Specifically, we define our attribute-based knowledge distillation loss as follows;

$$\mathcal{L}_{AKD} = \sum_{y} \sum_{k} \mathrm{D}^2 \left(P_y, P_{a_k,y} \right) , \qquad (2)$$

where D is a distance metric between two distributions. In our experiments, we use the Maximum Mean Discrepancy (MMD) [13], a powerful method for comparing distributions without relying on specific assumptions about their shapes. MMD works by evaluating the difference in average values across functions in a special space called a Reproducing Kernel Hilbert Space (RKHS). To facilitate this comparison, it employs a kernel, such as the Gaussian Radial Basis Function (RBF) in our context, to convert the distributions into the RKHS, thereby enabling easier analysis. We experiment with different distance metrics in Sec. 4.6. By optimizing $\mathcal{L}_{AKD}$, i.e., $\mathrm{D}^2(P_y, P_{a_k,y}) \rightarrow 0 \implies P_{a_k,y} \approx P_y$, the group features are encouraged to mirror their respective class distribution, allowing the model to learn representations that capture the unique characteristics of each class while washing out the attribute-specific features that could lead to biased learning. The final objective for `Debiasify`'s training is given as follows;

$$\mathcal{L}_{hybrid} = \mathcal{L}_{ACE} + \alpha \mathcal{L}_{AKD} + \mathcal{L}_{KL} , \qquad (3)$$

where $\mathcal{L}_{KL} = \mathrm{KL}(c_s, c_d)$ is the Kullback-Leibler divergence between the shallow and deep classifier logits, and α is a hyperparameter that controls the balance between classification accuracy and knowledge transfer.

4. Experiments

We conduct experiments to assess the performance of `Debiasify` across various benchmarks and compare it with state-of-the-art (SOTA) methods for bias mitigation. To ensure a fair comparison with other clustering-based debiasing methods, we adopt the experimental settings in the previous SOTA, CFix [7], when applicable.

4.1. Evaluation Benchmarks

We evaluate `Debiasify` using three datasets: CelebA [23], Waterbirds [32] and Fitzpatrick [14].

CelebA is a real-world dataset for face attribute recognition containing 202,599 celebrity face images, each annotated with 40 binary attributes. Following other works [7, 34], we designate Male as the bias attribute, and we diversify the target label by selecting other attributes that exhibit the strongest correlation with the bias.

Waterbirds is a synthesized dataset created by combining two other datasets to establish a strong correlation between bird types (waterbird, landbird) and their backgrounds (water, land). It consists of 4,795 training examples; 95% of them having matching bird types and backgrounds; e.g., waterbirds with water background, while the other do not.

Fitzpatrick is a well-known medical dataset for skin lesion analysis consisting of 16,012 clinical images with 3 class labels. Each image is annotated with a Fitzpatrick score representing the skin tone, which we designate as the bias attribute. The Fitzpatrick scale consists of six scores, ranging from 1 (the lightest skin tone) to 6 (the darkest skin tone). As in previous work [40], we group skin tones 1 to 3 into a category representing lighter skin tones, and skin tones 4 to 6 into a category representing darker skin tones. The dataset is imbalanced, with significantly more images of lighter skin tones compared to darker skin tones (11,060 vs. 4,952, respectively).

4.2. Evaluation Metrics

We evaluate the accuracy for each combination of target and bias attribute values $(y,\ a)$, reporting the results as average-group accuracy (unbiased accuracy) and worst-group accuracy [7, 34]. The results are averaged over three independent runs.

4.3. Baseline and Competitors

Baseline. We compare Debiasify against a vanilla-trained model, which does not incorporate any specific countermeasures for bias mitigation.

Competitors. We benchmark against several SOTA unsupervised debiasing methods: LfF [28], CFix [7], BPA [34], and George [36], where the latter three are clustering-based methods. Additionally, for an upper-bound performance comparison, we include GDRO [32], a method that optimizes worst-group performance over a distributionally robust uncertainty set using explicit bias supervision.

4.4. Implementation Details

We use a ResNet18 model, pretrained on ImageNet, as the backbone for all methods. ResNet18 consists of four module layers, and we select the layers at the end of the second module and the final module as our shallow and deep layers, respectively. Before clustering, we apply PCA for dimensionality reduction. For preprocessing, images are resized to 224x224 for CelebA and Fitzpatrick, and 256x256 for Waterbirds, with standard augmentations including cropping, flipping, and normalization. We use official data splits and train Debiasify with Adam (learning rate: 1×10^{-4}, batch size: 100, weight decay: 0.01) for 50 epochs. We set α to 0.1 and perform grid search for γ, yielding best results of 0.003-0.01 for the different labels in CelebA, 0.02 for Waterbirds, and 0.06 for Fitzpatrick.

4.5. Main Results

Qualitative Results. Table 1 (top) demonstrates that Debiasify outperforms all competitors, including the supervised GDRO and the previous unsupervised CFix, in most of the classification tasks on CelebA dataset; e.g., for the Wearing Necklace target, our method achieves the highest accuracy at 71.14%, surpassing CFix at 68.99% and GDRO at 62.89%. Similarly, for Pale Skin, our method leads with 92.77%, compared to CFix at 91.17% and GDRO at 90.55%. Moreover, the improvements in worst-group accuracy achieved by Debiasify are notable, as detailed in Table 1 (bottom). Our method demonstrates gains, over second best performing method, of approximately 10% for Wavy Hair, 9% for Brown Hair, 5% for Wearing Necklace, and 4% for Double Chin. This achievement is significant given that Debiasify does not specifically target worst-group accuracy, unlike methods such as George and GDRO, which explicitly optimize for this metric. These improvements in worst-group accuracy are achieved without compromising the performance across other groups, thereby ensuring a balanced enhancement, as reflected in the unbiased accuracy.

Additionally, experiments on Waterbirds in Table 2 confirm the effectiveness of our method even in a challenging controlled environment. We observe improvements in the worst-group performance, with gains of 1.15% and 2.49% compared to CFix and GDRO, respectively.

In the medical domain, Debiasify demonstrates superior performance on the Fitzpatrick dataset (Table 3), achieving the highest accuracies of 82.69% and 59.18%, surpassing the second-best method by 4.08% and 3.06% in terms of unbiased and worst-group accuracy, respectively.

Feature Space Visualization. In Figure 4, we present the t-SNE visualizations of feature embeddings generated by the baseline model (left) and our proposed model (right) on the CelebA dataset, specifically for the task of classifying the Blond Hair label. The embeddings are extracted from the penultimate layer of each model. In these plots, we focus solely on the negative examples (Blond Hair = False) to provide a clearer visualization of the data distribution. The colors represent the gender attribute, with blue indicating female and green indicating male. The baseline model's embeddings show a more segregated distribution based on gender, while our model's embeddings exhibit a more intermixed distribution of female and male samples within the same class. This intermixing indicates that Debiasify is effective at reducing the influence of the gender bias attribute, promoting a more debiased representation of the data.

Table 1. Performance evaluation of Debiasify against others on CelebA dataset. Cells in blue and green represent the best and second-best results, respectively. Improvement gain compared to the second-best method is given between brackets in red.

(a) Unbiased Accuracy (%)

	Unsupervised						Supervised
Target	**Baseline**	**LfF**	**George**	**BPA**	**CFix**	**Ours**	**GDRO**
Double Chin	64.61±0.82	68.47±0.22	76.23±0.11	82.92±0.54	85.13±0.30	**86.19±0.21** (+1.06)	83.19±1.11
Pale Skin	71.50±1.60	75.23±0.74	78.22±3.75	90.06±0.75	91.17±0.04	**92.77±1.38** (+1.60)	90.55±0.84
Wearing Necklace	55.04±0.59	57.21±0.76	58.79±0.10	68.96±0.12	68.99±1.19	**71.14±0.52** (+2.15)	62.89±3.69
Wearing Hat	93.53±0.37	94.81±0.15	95.72±0.71	96.80±0.26	**97.88±0.09**	97.65±0.37 (–0.23)	96.84±0.46
Big Lips	60.87±0.58	62.15±0.06	64.99±0.13	66.50±0.24	65.40±0.48	**67.73±0.44** (+1.23)	63.70±0.44
Bangs	89.04±0.47	89.04±0.50	92.62±0.12	93.94±0.57	94.67±0.16	**95.56±0.93** (+0.89)	94.45±0.17
Receding Hairline	69.72±0.78	74.58±0.21	78.86±0.40	84.95±0.49	**87.00±0.12**	86.84±0.33 (–0.16)	85.15±1.31
Wavy Hair	73.10±0.56	74.53±0.17	77.39±0.15	79.89±0.71	79.42±0.12	**80.80±1.26** (+1.38)	79.65±0.63
Brown Hair	78.07±0.87	78.93±1.24	83.07±0.07	83.83±0.66	85.30±0.47	**86.20±0.67** (+0.90)	84.87±0.07

(b) Worst-Group Accuracy (%)

	Unsupervised						Supervised
Target	**Baseline**	**LfF**	**George**	**BPA**	**CFix**	**Ours**	**GDRO**
Double Chin	21.33±2.24	28.24±0.46	50.00±0.41	67.78±0.91	74.26±3.94	**78.69±0.32** (+4.43)	72.94±1.14
Pale Skin	36.64±3.53	43.26±1.40	62.03±16.50	88.60±1.48	87.01±1.46	**89.76±0.62** (+1.16)	87.68±2.37
Wearing Necklace	02.72±0.83	06.67±2.07	13.82±0.41	41.93±2.47	55.56±0.38	**60.98±0.20** (+5.42)	24.34±7.81
Wearing Hat	85.12±0.31	88.31±0.12	92.93±0.76	94.94±0.19	96.58±0.63	**96.75±0.33** (+0.17)	94.67±0.41
Big Lips	30.85±0.62	38.54±0.18	44.51±0.83	56.99±3.05	57.27±0.58	**57.79±0.15** (+0.52)	47.55±1.03
Bangs	76.91±3.27	82.37±0.52	85.90±0.24	92.21±1.24	**93.01±0.36**	92.66±0.38 (–0.35)	92.12±1.03
Receding Hairline	35.69±0.35	45.53±0.55	57.30±0.90	79.11±1.91	84.15±0.82	**84.68±0.19** (+0.53)	79.12±2.11
Wavy Hair	38.01±0.85	45.24±0.83	53.17±0.43	65.74±1.13	69.92±0.38	**80.05±0.39** (+10.13)	66.79±1.62
Brown Hair	59.58±2.55	60.68±3.62	73.20±0.88	71.50±0.97	79.18±0.50	**88.10±0.28** (+8.92)	78.92±1.61

Table 2. Performance evaluation of Debiasify against others on Waterbirds dataset. Cells in blue and green represent the best and second-best results, respectively. Improvement gain compared to the second-best method is given between brackets in red.

Unbiased Accuracy (%)						**Worst-Group Accuracy (%)**					
Unsupervised					**Sup.**	**Unsupervised**					**Sup.**
Baseline	**LfF**	**BPA**	**CFix**	**Ours**	**GDRO**	**Baseline**	**LfF**	**BPA**	**CFix**	**Ours**	**GDRO**
87.99	85.05	88.44	**92.17**	91.49 (–0.68)	89.20	73.34	60.00	79.16	86.61	**87.76** (+1.15)	85.27

Table 3. Performance evaluation of Debiasify against others on Fitzpatrick dataset. Cells in blue and green represent the best and second-best results, respectively. Improvement gain compared to the second-best method is given between brackets in red.

Unbiased Accuracy (%)				**Worst-Group Accuracy (%)**			
Unsupervised			**Sup.**	**Unsupervised**			**Sup.**
Baseline	**LfF**	**Ours**	**GDRO**	**Baseline**	**LfF**	**Ours**	**GDRO**
77.90	78.61	**82.69** (+4.08)	77.12	36.73	42.17	**59.18** (+3.06)	56.12

Model Explainability. In Figure 5, we use Grad-CAM [33] to explore the interpretability of Debiasify for the Wearing Necklace, Wearing Hat, and Wavy Hair classification tasks in CelebA dataset. Our method consistently focuses on regions highly relevant to the target attribute, while the baseline often emphasizes unrelated bias-based features. For instance, in Wearing Necklace classification task, our model focuses on the necklace, whereas the baseline focuses on the mouth. We further demonstrate the effectiveness of our method on images from the worst-group category in the Waterbirds and Fitzpatrick datasets (Figure 5). In Waterbirds, our method clearly focuses on the bird, while the baseline emphasizes the background. Similarly, in Fitzpatrick, our method targets the specific skin lesion, whereas the baseline highlights the surrounding skin surface.

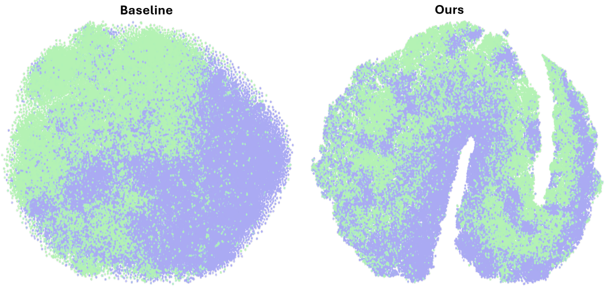
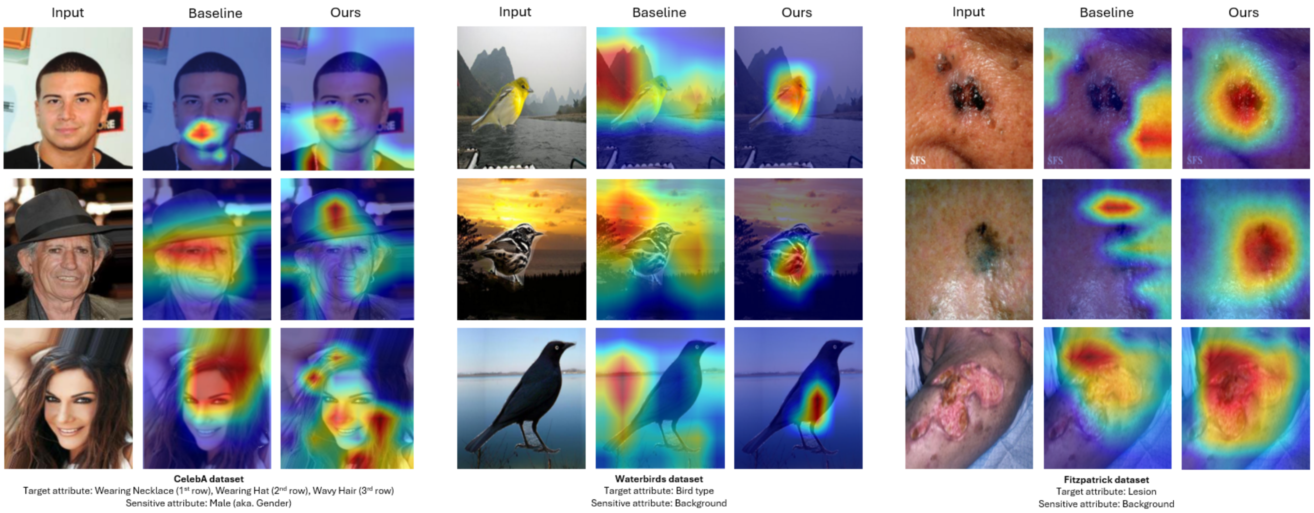


Figure 4. The t-SNE plots of feature embeddings for the baseline model (left) and our model (right) trained to classify Blond Hair. The plots display the distribution of samples with the target value Blond Hair = False. Blue and green colors represent female and male genders, respectively. Our Debiasify promotes a better mix of samples with the same target but different bias attribute values, which reduces the bias.

Multiple Bias Attributes. Thanks to the unsupervised design of Debiasify, we can seamlessly evaluate its performance in multi-bias scenarios without modifying the network or training framework. Table 4 shows the unbiased accuracy of Debiasify compared to other methods for the Blond Hair classification task when multiple bias attributes are present. Our method consistently achieves



CelebA dataset
Target attribute: Wearing Necklace (1st row), Wearing Hat (2nd row), Wavy Hair (3rd row)
Sensitive attribute: Male (aka. Gender)

Waterbirds dataset
Target attribute: Bird type
Sensitive attribute: Background

Fitzpatrick dataset
Target attribute: Lesion
Sensitive attribute: Background

Figure 5. Visualization of the class activation maps generated by GradCAM for the baseline and `Debiasify` (ours) on images from the CelebA (left), Waterbirds (middle), and Fitzpatrick (right) datasets.

Table 4. Performance evaluation (Unbiased Accuracy %) of `Debiasify` against others for `Blond Hair` classification with multiple bias attributes in the CelebA dataset. Cells in blue and green represent the best and second-best results, respectively. Improvement gain compared to the second-best method is given between brackets in red.

Method	Bias	`Male, Young`	`Male, Big Nose`	`Male, Smiling`	`Male, Heavy Makeup`	`Male, Wearing Lipstick`	`Male, Wearing Necklace`
Unsup.	**Baseline**	78.39	81.18	79.75	83.64	80.34	79.25
	LfF	81.21	84.10	82.91	**88.82**	84.13	81.03
	Ours	**88.35** (+0.39)	**91.24** (+0.41)	89.78 (–1.95)	88.71 (–0.11)	**87.21** (+1.28)	90.96 (–1.30)
Sup.	**GDRO**	87.96	90.83	**91.73**	81.09	85.93	**92.26**

superior results across various bias sets compared to other unsupervised methods. Additionally, it performs close to the supervised GDRO, which is very sensitive to the bias sets, as it trains a separate model for each bias. In contrast, `Debiasify` can be applied to any bias set without further fine-tuning.

Linear Decodability. As discussed in Sec 3.1, we repeat the linear decodability experiments using our `Debiasify`. Figure 3 shows that bias decodability, whether for a single or combined attributes, is consistently lower with our method (orange bars) compared to the baseline (blue bars), demonstrating its effectiveness in mitigating bias across all layers.

4.6. Ablation Studies

We perform five sensitivity analyses on the CelebA (`Double Chin` and `Wearing Necklace`), Waterbirds, and Fitzpatrick datasets, summarized in Table 5.

1. Cluster Assignment. In Exp. $\mathcal{A}$, we evaluate the impact of using true bias distributions versus pseudo clusters derived from the shallow layer on the performance of `Debiasify`. Specifically, we replace the attribute-conditioned clusters $P_{a_k,y}$ in Eq. 2 with the true bias distributions, assuming they are available during training. The results demonstrate that utilizing our pseudo clusters derived from the shallow layer is generally effective, with performance comparable to or marginally better than using true bias distributions in Exp. $\mathcal{A}$. Particularly notable improvements are observed in the Fitzpatrick dataset, where `Debiasify` achieves a higher accuracy, especially in worst-group performance, compared to using true bias distributions. A possible explanation for this result lies in the inherent noise or imperfections in the true bias labels. Clustering through `Debiasify` allows for a more nuanced understanding and adaptation to subtle variations within the dataset that may not be fully captured by the explicit bias labels, leading to improved performance.

2. Distillation Depth. We assess how different distillation depths impact `Debiasify` by fixing the deep layer at the default position (layer 4) and systematically adjusting the shallow layer involved in the distillation process. Specifically, we evaluate the following shallow layer configurations: Layer 1 (Exp. $\mathcal{B}$), layer 3 (Exp. $\mathcal{C}$), the combination of layers 1 and 2 (Exp. $\mathcal{D}$), and the combination of layers 1, 2, and 3 (Exp. $\mathcal{E}$). Our findings are as follows: Interestingly, our method is effective across different shallow layer configurations; i.e., distilling knowledge from the deep layer into any shallow layer consistently improves de-

Table 5. Performance evaluation, given as unbiased (worst-group) accuracy, of Debiasify from different ablation studies on CelebA, Waterbirds and Fitzpatrick datasets.

	CelebA (`Double Chin`)	**CelebA** (`Wearing Necklace`)	**Waterbirds**	**Fitzpatrick**
Default / **Exp.**	86.19 (78.69)	71.14 (60.98)	91.49 (87.76)	82.69 (59.18)
$\mathcal{A}$	84.37 (80.15)	71.34 (61.87)	91.16 (88.32)	81.36 (54.64)
$\mathcal{B}$	84.26 (78.13)	68.36 (61.75)	90.26 (85.36)	81.24 (57.83)
$\mathcal{C}$	84.37 (77.62)	69.22 (59.17)	91.67 (86.42)	80.65 (58.27)
$\mathcal{D}$	83.82 (78.45)	68.29 (59.54)	89.28 (86.19)	81.44 (58.67)
$\mathcal{E}$	85.11 (80.62)	71.88 (62.30)	90.85 (87.44)	83.09 (60.31)
$\mathcal{F}$	83.82 (69.83)	70.32 (56.28)	88.65 (83.21)	81.67 (57.24)
$\mathcal{G}$	84.61 (73.29)	69.88 (58.39)	89.27 (84.69)	82.35 (58.61)
$\mathcal{H}$	83.82 (69.85)	70.93 (56.36)	89.54 (83.29)	80.35 (46.54)
$\mathcal{I}$	83.45 (78.68)	70.32 (59.11)	90.73 (84.68)	82.11 (57.29)
$\mathcal{J}$	84.00 (72.06)	68.74 (56.37)	90.23 (85.41)	81.68 (58.67)
$\mathcal{L}$	83.63 (73.53)	69.35 (59.21)	91.16 (86.32)	82.35 (59.27)
$\mathcal{M}$	85.17 (83.09)	69.21 (63.48)	91.25 (86.27)	82.24 (58.91)

biasing, leading to superior worst-group accuracy compared to all SOTA methods across all datasets (Tables 1, 2, 3). This demonstrates the strong impact of our proposed deep-to-shallow distillation mechanism. 2) The multi-layer distillation in Exp. $\mathcal{E}$ achieves the highest worst-group accuracy across the other configurations, which is expected as it guides all layers to learn debiased representations. However, it requires longer training and increased complexity due to the multi-layer clustering and distillation. 3) Based on average results across datasets, the default Debiasify configuration (i.e., layer 2) offers the best performance balance while avoiding the computational overhead of Exp. $\mathcal{E}$.

3. Distance Metric Selection. To assess the sensitivity of Debiasify to the choice of distance metric used for the learning of debiased representations, we replace MMD in Eq. 2 with Kullback-Leibler (KL) divergence (Exp. $\mathcal{F}$) and Mahalanobis distance (Exp. $\mathcal{G}$). We observe that while both alternative metrics show a slight decline in unbiased accuracy, the use of Mahalanobis distance (Exp.$\mathcal{G}$) results in a relatively smaller decrease in worst-group accuracy compared to KL divergence (Exp. $\mathcal{F}$). Overall, the results suggest that MMD is the most effective distance metric for our method, providing the best balance between unbiased accuracy and worst-group performance.

4. Hybrid Loss Components. In Exp. $\mathcal{H}$ and $\mathcal{I}$, we evaluate the impact of omitting different components of the hybrid loss $\mathcal{L}_{hybrid}$ (Eq. 3). We notice that omitting $\mathcal{L}_{AKD}$ (Exp. $\mathcal{H}$) significantly decreases worst-group accuracy across all datasets, underscoring its crucial role in learning debiased representations. Conversely, omitting $\mathcal{L}_{KL}$ (Exp. $\mathcal{I}$) results in less performance degradation, as this component primarily aids in knowledge transfer between logits, which is less essential for developing robust, debiased features.

5. Number of Clusters. We investigate the performance of Debiasify by varying the threshold value γ to obtain different number of clusters per class. Specifically, we experiment with $\mathcal{K} = 2$, 4 and 16 in Exps. $\mathcal{J}$, $\mathcal{L}$, and $\mathcal{M}$, respectively. Note that the default values of γ reported in Sec. 4.4 result in $\mathcal{K}$ = 8 for the CelebA and Waterbirds datasets, and $\mathcal{K}$ = 2 for the Fitzpatrick dataset. We observe that using a higher number of clusters ($\mathcal{K} = 16$ in Exp. $\mathcal{M}$) generally improves the worst-group accuracy across the datasets. However, this comes at the cost of decreased unbiased accuracy and increased computational complexity and training time. Conversely, fewer clusters ($\mathcal{K} = 2$ in Exp. $\mathcal{J}$) result in a notable drop in performance in the CelebA dataset, particularly in worst-group accuracy, while having less impact on the Waterbirds and Fitzpatrick datasets. This suggests that a moderate number of clusters is sufficient to capture the necessary attribute-conditioned variations while maintaining the model's overall robustness and efficiency.

5. Conclusions

We present a robust unsupervised debiasing framework that leverages feature clustering and self-distillation. Debiasify is based on the observation that images with similar non-target attribute labels cluster prominently in shallow neural network layers. Building on this, we introduce a novel clustering-based method, Debiasify, which leverages these attribute-conditioned clusters to learn debiased representations using a self-distillation technique. Our technique enforces the distributions of these clusters to converge towards each other while simultaneously aligning them with the distribution of their respective class in the deepest layer, where more complex and predictive features reside. We demonstrate Debiasify's effectiveness through extensive experiments, outperforming previous debiasing methods, particularly in worst-group accuracy.

Future Work. While Debiasify has proven robust across different shallow layer configurations, future work could develop a systematic approach or trainable module to automatically select the optimal layer(s) for enhanced debiasing, improving adaptability across a wider range of architectures and datasets.

References

[1] Saeid Asgari, Aliasghar Khani, Fereshte Khani, Ali Gholami, Linh Tran, Ali Mahdavi Amiri, and Ghassan Hamarneh. Masktune: Mitigating spurious correlations by forcing to explore. *Advances in Neural Information Processing Systems*, 35:23284–23296, 2022. 2

[2] Haoyue Bai, Fengwei Zhou, Lanqing Hong, Nanyang Ye, S-H Gary Chan, and Zhenguo Li. NAS-OOD: Neural architecture search for out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8320–8329, 2021. 1

[3] Nourhan Bayasi, Jamil Fayyad, Alceu Bissoto, Ghassan Hamarneh, and Rafeef Garbi. Biaspruner: Debiased continual learning for medical image classification. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pages 90–101, 2024. 1, 2

[4] Nourhan Bayasi, Ghassan Hamarneh, and Rafeef Garbi. Boosternet: Improving domain generalization of deep neural nets using culpability-ranked features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 538–548, 2022. 1

[5] Yakir Berchenko. Simplicity bias in overparameterized machine learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 11052–11060, 2024. 2

[6] Sumon Biswas and Hridesh Rajan. Fair preprocessing: towards understanding compositional fairness of data transformers in machine learning pipeline. In *Proceedings of the 29th ACM joint meeting on European software engineering conference and symposium on the foundations of software engineering*, pages 981–993, 2021. 2

[7] Giacomo Capitani, Federico Bolelli, Angelo Porrello, Simone Calderara, and Elisa Ficarra. Clusterfix: A cluster-based debiasing approach without protected-group supervision. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4870–4879, 2024. 2, 3, 4, 5

[8] Ching-Yao Chuang and Youssef Mroueh. Fair mixup: Fairness via interpolation. In *International Conference on Learning Representations*, 2021. 2

[9] Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment inference for invariant learning. In *International Conference on Machine Learning*, pages 2189–2200, 2021. 1

[10] Nikolay Dagaev, Brett D Roads, Xiaoliang Luo, Daniel N Barry, Kaustubh R Patil, and Bradley C Love. A too-good-to-be-true prior to reduce shortcut reliance. *Pattern Recognition Letters*, 166:164–171, 2023. 1, 3

[11] Siyi Du, Ben Hers, Nourhan Bayasi, Ghassan Hamarneh, and Rafeef Garbi. FairDisCo: Fairer ai in dermatology via disentanglement contrastive learning. In *European Conference on Computer Vision*, pages 185–202, 2022. 1, 2

[12] Thomas Duboudin, Emmanuel Dellandréa, Corentin Abgrall, Gilles Hénaff, and Liming Chen. Look beyond bias with entropic adversarial data augmentation. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 2142–2148, 2022. 1

[13] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012. 4

[14] Matthew Groh, Caleb Harris, Luis Soenksen, Felix Lau, Rachel Han, Aerin Kim, Arash Koochek, and Omar Badri. Evaluating deep neural networks trained on clinical images in dermatology with the fitzpatrick 17k dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1820–1828, 2021. 4

[15] Katherine Hermann and Andrew Lampinen. What shapes feature representations? exploring datasets, architectures, and training. *Advances in Neural Information Processing Systems*, 33:9995–10006, 2020. 3

[16] Inwoo Hwang, Sangjun Lee, Yunhyeok Kwak, Seong Joon Oh, Damien Teney, Jin-Hwa Kim, and Byoung-Tak Zhang. Selecmix: Debiased learning by contradicting-pair sampling. *Advances in Neural Information Processing Systems*, 35:14345–14357, 2022. 2

[17] Sangwon Jung, Donggyu Lee, Taeeon Park, and Taesup Moon. Fair feature distillation for visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12115–12124, 2021. 1

[18] Eungyeup Kim, Jihyeon Lee, and Jaegul Choo. Biaswap: Removing dataset bias with bias-tailored swapping augmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14992–15001, 2021. 2

[19] Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. Last layer re-training is sufficient for robustness to spurious correlations. *arXiv preprint arXiv:2204.02937*, 2022. 1

[20] Alina Köchling and Marius Claus Wehner. Discriminated by an algorithm: a systematic review of discrimination and fairness by algorithmic decision-making in the context of hr recruitment and hr development. *Business Research*, 13(3):795–848, 2020. 1

[21] Jungsoo Lee, Eungyeup Kim, Juyoung Lee, Jihyeon Lee, and Jaegul Choo. Learning debiased representation via disentangled feature augmentation. *Advances in Neural Information Processing Systems*, 34:25123–25133, 2021. 2

[22] Evan Z Liu, Behzad Haghgoo, Annie S Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. Just train twice: Improving group robustness without training group information. In *International Conference on Machine Learning*, pages 6781–6792, 2021. 1, 2

[23] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015. 4

[24] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15(2018):11, 2018. 1

[25] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982. 4

[26] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning adversarially fair and transferable representations. In *International Conference on Machine Learning*, pages 3384–3393, 2018. 2

[27] Rémi Nahon, Van-Tam Nguyen, and Enzo Tartaglione. Mining bias-target alignment from voronoi cells. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4946–4955, 2023. 2

[28] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33:20673–20684, 2020. 1, 2, 5

[29] Hongjing Niu, Hanting Li, Feng Zhao, and Bin Li. Roadblocks for temporarily disabling shortcuts and learning new knowledge. *Advances in Neural Information Processing Systems*, 35:29064–29075, 2022. 1

[30] Felix Petersen, Debarghya Mukherjee, Yuekai Sun, and Mikhail Yurochkin. Post-processing for individual fairness. *Advances in Neural Information Processing Systems*, 34:25944–25955, 2021. 2

[31] Mohammad Pezeshki, Oumar Kaba, Yoshua Bengio, Aaron C Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *Advances in Neural Information Processing Systems*, 34:1256–1272, 2021. 1

[32] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019. 2, 4, 5

[33] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-CAM: visual explanations from deep networks via gradient-based localization. *International journal of computer vision*, 128:336–359, 2020. 6

[34] Seonguk Seo, Joon-Young Lee, and Bohyung Han. Unsupervised learning of debiased representations with pseudo-attributes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16742–16751, 2022. 2, 3, 4, 5

[35] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems*, 33:9573–9585, 2020. 1, 2

[36] Nimit Sohoni, Jared Dunnmon, Geoffrey Angus, Albert Gu, and Christopher Ré. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *Advances in Neural Information Processing Systems*, 33:19339–19352, 2020. 2, 5

[37] Rishabh Tiwari and Pradeep Shenoy. Overcoming simplicity bias in deep networks using a feature sieve. In *International Conference on Machine Learning*, pages 34330–34343, 2023. 1, 2, 3

[38] Rishabh Tiwari, Durga Sivasubramanian, Anmol Mekala, Ganesh Ramakrishnan, and Pradeep Shenoy. Using early readouts to mediate featural bias in distillation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2638–2647, 2024. 3

[39] Shirley Wu, Mert Yuksekgonul, Linjun Zhang, and James Zou. Discover and cure: Concept-aware mitigation of spurious correlation. In *International Conference on Machine Learning*, pages 37765–37786. PMLR, 2023. 2

[40] Yawen Wu, Dewen Zeng, Xiaowei Xu, Yiyu Shi, and Jingtong Hu. Fairprune: Achieving fairness through pruning for dermatological disease diagnosis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 743–753. Springer, 2022. 5

[41] Ruicheng Xian, Lang Yin, and Han Zhao. Fair and optimal classification via post-processing. In *International Conference on Machine Learning*, pages 37977–38012, 2023. 2

[42] Zeliang Zhang, Mingqian Feng, Zhiheng Li, and Chenliang Xu. Discover and mitigate multiple biased subgroups in image classifiers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10906–10915, 2024. 2