BAdd: Bias Mitigation through Bias Addition

Ioannis Sarridis[1,2] Christos Koutlis[1] Symeon Papadopoulos[1] Christos Diou[2]
[1]Information Technologies Institute, CERTH, Greece
[2]Department of Informatics and Telematics, Harokopio University of Athens, Greece
{gsarridis, ckoutlis, papadop}@iti.gr {isarridis, cdiou}@hua.gr

Abstract

Computer vision datasets often exhibit biases in the form of spurious correlations between certain attributes and target variables. While recent efforts aim to mitigate such biases and foster bias-neutral representations, they fail in complex real-world scenarios. In particular, existing methods excel in controlled experiments on benchmarks with single-attribute injected biases, but struggle with complex multi-attribute biases that naturally occur in established CV datasets. In this paper, we introduce BAdd, a simple yet effective method that allows for learning bias-neutral representations invariant to bias-inducing attributes. This is achieved by injecting features encoding these attributes into the training process. BAdd is evaluated on seven benchmarks and exhibits competitive performance, surpassing state-of-the-art methods on both single- and multi-attribute bias settings. Notably, it achieves +27.5% and +5.5% absolute accuracy improvements on the challenging multi-attribute benchmarks, FB-Biased-MNIST and CelebA, respectively.

1. Introduction

Deep Learning (DL) models have demonstrated impressive capabilities and groundbreaking performance across various Computer Vision (CV) tasks [11, 13, 38]. However, a concerning issue has emerged alongside these advancements: the potential for bias in Artificial Intelligence (AI) systems, disproportionately impacting specific groups [6, 12, 31]. Specifically, when AI systems base their decisions, often indirectly, on attributes like age, gender, or race, they become discriminatory. Considering the profound impact AI decisions can have on individuals' lives, such biases should be mitigated before deployment in high-stakes applications [7, 10, 37, 38]. Moreover, even when such biases are not demographic-related but stem from "shortcuts" that prioritize irrelevant features, addressing them is crucial for building more robust and reliable CV systems [22, 28].

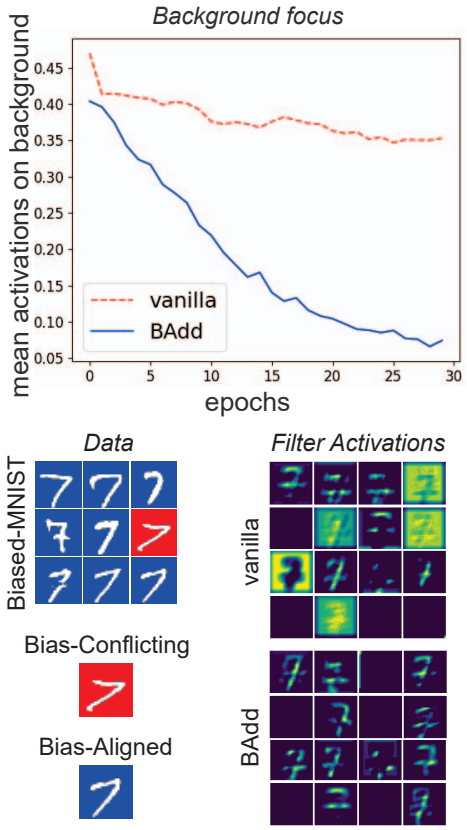


Figure 1. During training on Biased-MNIST, where the color-digit association is strong, a vanilla model struggles with bias, as reducing reliance on the protected attribute (here 'color') results in increased loss for samples that deviate from this spurious correlation. In contrast, BAdd results in learning bias-neutral feature representations of the digits, independent of color. This is evidenced by the activation maps on the samples where bias occurs.

Bias in CV often originates from the composition of training datasets [12]. The most common way bias arises in training sets is through a data selection process that associates specific groups of people or objects with certain

visual attributes. When such data is used to train DL models, the attribute associations lead the model to prioritize irrelevant attributes in its decision making process [27, 45]. Motivated by this issue, several approaches have been proposed to enable learning bias-neutral representations that are robust to the so-called *bias attributes* [5, 16, 30], i.e., attributes that exhibit spurious correlations with the target classes. Such methods often leverage labels associated with protected attributes to guide model training towards learning bias-neutral representations [4, 5, 8, 9, 16, 30] through techniques like adversarial training [17, 41] and regularization [5, 16, 30, 39].

A fundamental limitation of existing approaches lies in their loss-based nature. Typically, these methods introduce additional loss terms to penalize the biased model's behavior, which retroactively corrects bias that is already introduced in the model's learning process. While methods adopting this strategy may appear sound in theory and demonstrate state-of-the-art performance on simple datasets, they struggle with more complex forms of bias, especially when dealing with multiple biased attributes, and demonstrate sub-optimal performance. To overcome these challenges, there is a need for more proactive bias mitigation approaches that intervene earlier in the training process to address the root cause of bias propagation in the model itself. By disrupting the process through which bias enters the model, we can build models that are more effective for a wide range of complex biases present in CV datasets.

In this paper, we introduce BAdd, a simple yet effective and versatile in-processing bias mitigation method. The proposed method relies on the principle that injecting *bias-capturing features* into the penultimate layer's output enables learning representations invariant to these features (see Fig. 1). Deriving bias-capturing features is straightforward in practice, since it can be formulated as the task of predicting the values of biased attributes. BAdd intervenes in the mechanism by which bias is introduced to DL models during training via the minimization of the loss function. In particular, a vanilla model optimizes its parameters by taking advantage of biases present in the data, as doing so reduces the overall loss. Such a model learns to prioritize features associated with the biased attributes, reinforcing and perpetuating the bias within its representations. To alleviate this issue, BAdd suggests that the intentional inclusion of bias-capturing features in the training process ensures that the attributes introducing the bias do not exert undue influence on the loss function optimization, and thus the trainable parameters of the model are not affected by them. In essence, BAdd decouples the learning of biased features from the optimization process and thus allows for learning bias-neutral representations. BAdd outperforms or is on par with state-of-the-art bias mitigation methods on a wide range of experiments involving four datasets with single attribute biases (i.e., Biased-MNIST, Biased-UTKFace, Waterbirds, and Corrupted-CIFAR10) and three datasets with multi-attribute biases (i.e., FB-Biased-MNIST, UrbanCars, and CelebA). Where BAdd shines is on datasets with multi-attribute biases, where it outperforms the state of the art by +27.5%, and +5.5% absolute accuracy improvements on FB-Biased-MNIST, and CelebA, respectively.

In summary, the paper makes the following contributions: (i) we introduce BAdd, an effective methodology for learning bias-neutral representations concerning one or more protected attributes by incorporating bias-capturing features into the model's representations (ii) we provide an extensive evaluation involving seven benchmarks, demonstrating the superiority of BAdd on both single- and multi-attribute bias scenarios. BAdd implementation is available as part of the VB-Mitigator library [29].

2. Related Work

Bias-aware image classification benchmarks. Most standard benchmarks for evaluating bias mitigation methods in CV involve artificially generated single-attribute biases. Biased-MNIST [4], a MNIST derivative dataset, associates each digit with a specific colored background. Similarly, Corrupted-CIFAR10 [15] introduces biased textures across the classes of CIFAR10. The Waterbirds [28] dataset is constructed by cropping birds from the CUB-200 [40] dataset and transferring them onto backgrounds from the Places dataset [48], introducing correlations between bird species and certain backgrounds (i.e., habitat types). On the other hand, datasets like Biased-UTKFace [16] and Biased-CelebA [16] are carefully selected subsets of UTKFace [47] and CelebA [24], respectively, designed to exhibit an association of 90% between specific attributes, such as `gender` and `race`. Despite their value in research, all these benchmarks share a crucial limitation: they are far from capturing the complexities of real-world dataset biases, as they typically exhibit uniformly distributed single attribute biases. To address this limitation, recent works introduced benchmarks that involve multi-attribute biases, such as Biased-MNIST variations [2, 33, 34] and UrbanCars [22]. The latter introduces a multi-attribute bias setting by incorporating biases related to both background and co-occurring objects and the task is to classify the car body type into `urban` or `country` car.

In addition to the above benchmarks, in this paper, we create a variation of Biased-MNIST, termed FB-Biased-MNIST, which builds on the background color bias in Biased-MNIST by injecting an additional foreground color bias. Furthermore, we consider a benchmark that utilizes the original, unmodified CelebA dataset but focuses on evaluating performance against the most prominent bias-inducing attributes in the dataset. This allows for evaluating bias-aware methods on multiple biases in a more realistic

setting, without artificially enforced biases.

Bias-aware approaches. Efforts on learning bias-neutral representations using biased data encompass techniques like ensemble learning [9, 42], contrastive learning [5, 16], adversarial frameworks [1, 3, 17, 36, 41, 44], and regularization approaches [8, 16, 30, 39]. For instance, the Learning Not to Learn (LNL) approach [17] penalizes models if they predict protected attributes, while the Domain-Independent (DI) approach [42] introduces the usage of domain-specific classifiers to mitigate bias. Entangling and Disentangling deep representations (EnD) [39] suggests a regularization term that entangles or disentangles feature vectors w.r.t. their target and protected attribute labels. FairKL [5] and BiasContrastive-BiasBalance (BC-BB) [16] employ contrastive learning for bias mitigation by utilizing the pairwise similarities of samples in the feature space. Finally, there are several works that can be employed without utilizing the protected attribute labels, such as Learned-Mixin (LM) [9], Rubi [8], ReBias [4], Learning from Failure (LfF) [25], and FLAC [30]. The latter achieves state-of-the-art performance by utilizing a bias-capturing classifier and a sampling strategy that effectively focuses on the underrepresented groups.

It is worth noting that methodologies for distributionally robust optimization [21–23, 26, 28, 43] are relevant to the field of bias mitigation, as they aim at mitigating biases arising from spurious correlations in the training data. Similarly to the above bias mitigation methods, [28] and [21] suggest regularization terms to mitigate such correlations, while [23] and [43] introduced methods that try to compensate the effect of spurious correlations by increasing or decreasing the weights of certain training samples. Based on the same idea, [26] focuses on reweighting the features rather than the samples. Finally, the Last Layer Ensemble (LLE) [22] employs multiple augmentations to eliminate different biases (i.e., one type of augmentation for each type of bias). However, LLE requires extensive pre-processing (e.g., object segmentation), which makes it challenging to apply or even infeasible in new CV datasets. On the other hand, BAdd is a simple yet effective approach that can be easily applied to any network architecture and CV dataset.

3. Bias Mitigation

3.1. Problem Formulation

Consider a dataset $\mathcal{D}$ comprising training samples $(\mathbf{x}^{(i)}, y^{(i)})$, where $\mathbf{x}^{(i)}$ represents the input sample and $y^{(i)}$ belongs to the set of target labels $\mathcal{Y}$. Let $h(\cdot)$ denote a model trained on $\mathcal{D}$ and $\mathbf{h}$ the model feature representation (e.g., output of penultimate model layer). Let also $\mathcal{T}$ be the domain of tuples of *protected attributes*, e.g., $t = (male, 25, black) \in \mathcal{T}$ for protected attributes `gender`, `age` and `race`. The objective is to train h such that the protected attributes are not used to predict the targets in $\mathcal{Y}$. In addition, we also assume that a bias-capturing model $b(\cdot)$, with feature representation $\mathbf{b}$, has been trained to predict the value of the protected attribute(s) $t \in \mathcal{T}$ from $\mathbf{x}$.

We define $\mathcal{D}$ as *biased* with respect to the protected attributes in $\mathcal{T}$ if there is a high correlation of certain values in $\mathcal{Y}$ with a value or a combination of values of protected attributes in $\mathcal{T}$. Within a batch $\mathcal{B}$, samples exhibiting the dataset bias are termed *bias-aligned* ($\mathcal{B}_\mathcal{A}$), while those that deviate from it are referred to as *bias-conflicting* ($\mathcal{B}_\mathcal{C}$). The set $\mathcal{D}$ is assumed to include at least some bias-conflicting examples. Note that bias-aligned and bias-conflicting samples correspond to the over-represented and under-represented groups within $\mathcal{D}$, respectively. Using such a biased dataset for training often introduces model bias, by leading h to encode information related to t. Our objective is to mitigate these dependencies between representations $\mathbf{h}$ and $\mathbf{b}$, leading to a bias-neutral feature representation.

3.2. The Vicious Circle of Bias

Before introducing the proposed methodology, we formally describe how bias is typically introduced in a vanilla model. When training a classification model $h(\cdot)$ on a biased dataset $\mathcal{D}$, the model often prioritizes learning features correlated with protected attributes rather than those that directly characterize the target class. This phenomenon arises in cases of high correlation between protected attributes and targets, provided that the protected attribute's visual characteristics are easier to capture than the visual characteristics of the target [45]. Below, we delve into the details behind a vanilla model's inherent inclination towards this kind of bias and explain how the proposed approach addresses this limitation.

First, let us consider the Cross-Entropy loss on a batch of samples $\mathcal{B} = \mathcal{B}_\mathcal{A} \cup \mathcal{B}_\mathcal{C}$:

$$\mathcal{L} = -\frac{1}{N}\sum_{i\in\mathcal{B}}\sum_{k=1}^{K} y_k^{(i)} \log \hat{y}_k^{(i)} = -\frac{1}{N}\sum_{i\in\mathcal{B}_\mathcal{A}}\sum_{k=1}^{K} y_k^{(i)} \log \hat{y}_k^{(i)} - \frac{1}{N}\sum_{i\in\mathcal{B}_\mathcal{C}}\sum_{k=1}^{K} y_k^{(i)} \log \hat{y}_k^{(i)} = \mathcal{L}_{\mathcal{B}_\mathcal{A}} + \mathcal{L}_{\mathcal{B}_\mathcal{C}}. \tag{1}$$

where N is the number of samples within a batch, and K is the number of target classes. The predictions $\hat{y}_k^{(j)}$ are computed via multinomial logistic regression, as follows:

$$\hat{y}_k^{(j)} = \sigma_k(\mathbf{z}(\mathbf{x}^{(j)}; \boldsymbol{\theta_h})), \tag{2}$$

where j is the index of input sample $\mathbf{x}^{(j)}$, $\boldsymbol{\theta_h}$ the learnable parameters of model $h(\cdot)$ and σ_k the k-th class prob-

ability after applying the softmax function on the logits $\mathbf{z}(\mathbf{x}^{(j)}; \boldsymbol{\theta_h})$.

Given that $||\mathcal{B}_\mathcal{A}|| >> ||\mathcal{B}_\mathcal{C}||$, where $||\cdot||$ denotes the cardinality of a set, we can assume that there exists a point in the training process at which the model has learned to be accurate on the bias-aligned samples $\mathcal{B}_\mathcal{A}$ misguidedly relying on protected attributes' features, so that $\mathcal{L}_{\mathcal{B}_\mathcal{A}} \approx 0$, while at the same time $\mathcal{L}_{\mathcal{B}_\mathcal{C}} >> 0$. Consequently, backpropagating the gradients of $\mathcal{L}$ will update the parameters $\boldsymbol{\theta_h}$ in a way that steers the model towards accurately predicting the bias-conflicting samples $\mathcal{B}_\mathcal{C}$ to further reduce $\mathcal{L}$, which stops reliance of $h(\cdot)$ on the protected attributes. The major limitation of a vanilla model is directly connected to the loss behavior when the model processes the next mini-batch. In particular, the step the model makes towards correctly predicting the samples in $\mathcal{B}_\mathcal{C}$, thus reducing $\mathcal{L}_{\mathcal{B}_\mathcal{C}}$, adversely affects the loss w.r.t. the bias-aligned samples, which is now $\mathcal{L}_{\mathcal{B}_\mathcal{A}} >> 0$, as $h(\cdot)$ relies less on the protected attributes and at the same time it is impossible to learn to encode the target with only one batch of $||\mathcal{B}_\mathcal{C}||$ bias-conflicting samples. This leads to a loss spike for the bias-aligned samples that in the next iteration will restore the model's parameters $\boldsymbol{\theta_h}$ to their initial state (encoding the protected attributes' features) in order to again achieve a much lower $\mathcal{L}$. Figure 2 illustrates this behavior through a snapshot of the losses and the gradients related to the bias-aligned and bias-conflicting samples during several training steps of the vanilla model (refer to the supplementary material for the BAdd model behavior). In this example, to emphasize the described phenomenon, we primarily include bias-aligned samples, with only 2 batches of bias-conflicting samples introduced every 200 training steps.

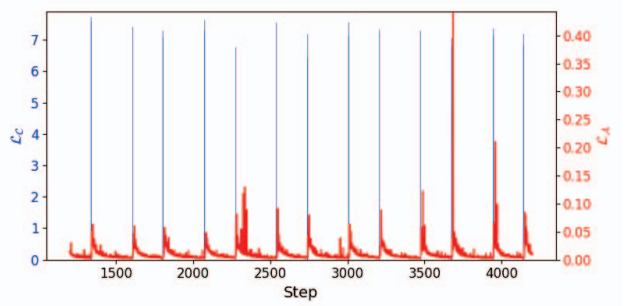
To better expose this behavior, let us consider the derivative of the loss of (1) w.r.t. a parameter θ_h^0 for the i-th sample:

$$\begin{aligned}\frac{\partial \mathcal{L}^{(i)}}{\partial \theta_h^0} &= y_\kappa \frac{\partial \log \sigma_\kappa(\mathbf{z}(\mathbf{x}^{(i)}; \boldsymbol{\theta_h}))}{\partial \theta_h^0} \\ &= y_\kappa \frac{1}{\sigma_\kappa(\mathbf{z}(\mathbf{x}^{(i)}; \boldsymbol{\theta_h}))} \frac{\partial \sigma_\kappa(\mathbf{z}(\mathbf{x}^{(i)}; \boldsymbol{\theta_h}))}{\partial \theta_h^0}.\end{aligned} \tag{3}$$

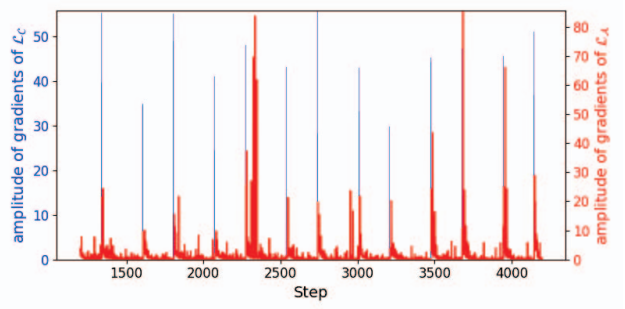
where κ is the correct class, according to the ground truth (i.e., $y_\kappa = 1$). Setting $A_0^{(i)} = \frac{\partial \sigma_\kappa(\mathbf{z}(\mathbf{x}^{(i)}; \boldsymbol{\theta_h}))}{\partial \theta_h^0}$ and $\sigma_\kappa^{(i)} = \sigma_\kappa(\mathbf{z}(\mathbf{x}^{(i)}; \boldsymbol{\theta_h}))$, the derivative for a batch becomes

$$\frac{\partial \mathcal{L}}{\partial \theta_h^0} = -\frac{1}{N}\Big(\sum_{i:\mathbf{x}^{(i)} \in \mathcal{B}_\mathcal{A}} \frac{1}{\sigma_\kappa^{(i)}} A_0^{(i)} + \sum_{j:\mathbf{x}^{(j)} \in \mathcal{B}_\mathcal{C}} \frac{1}{\sigma_\kappa^{(j)}} A_0^{(j)} \Big). \tag{4}$$

After the model has learned to predict the targets based on the protected attributes, $\sigma_\kappa^{(i)}$ is large (close to 1) while $A_0^{(i)}$ is small, as $h(\cdot)$ already correctly predicts samples in $\mathcal{B}_\mathcal{A}$. In contrast, $\sigma_\kappa^{(j)}$ is small while $A_0^{(j)}$ is large. The model update, therefore, strongly depends on the samples in $\mathcal{B}_\mathcal{C}$. After the update step, however, $\sigma_\kappa^{(i)}$ becomes smaller, $A_0^{(i)}$ becomes larger and given that $||\mathcal{B}_\mathcal{A}|| >> ||\mathcal{B}_\mathcal{C}||$, the derivative is now dominated by samples in $\mathcal{B}_\mathcal{A}$, and the parameters revert back to their previous values. In other words, any progress the model makes towards reducing its bias is counteracted by the loss function, which is lower when the model focuses on the easier-to-learn, biased samples. This essentially traps the model in a vicious circle where the model is condemned to encode the protected attributes instead of the targets.



(a) Losses



(b) Gradients

Figure 2. Biased-MNIST bias-conflicting samples trigger spikes on $\mathcal{L}_\mathcal{A}$ and gradients of $\mathcal{L}_\mathcal{A}$. Blue bars indicate the steps where bias-conflicting samples occur, with height representing y-axis values.

3.3. Bias Mitigation through Bias Addition

BAdd proposes incorporating the features $\mathbf{b}$ that capture the protected attributes of the dataset in the model's feature representation $\mathbf{h}$. Feature representation $\mathbf{b}$ encapsulates all the desired protected attributes and can be considered as $\mathbf{b} = \mathbf{b}_1 + \mathbf{b}_2 + \cdots + \mathbf{b}_M$ where $M = |\mathcal{T}|$ is the number of protected attributes in the dataset. These features can be obtained either by training a bias-capturing classifier or, in case the protected attribute labels are known, by projecting them into the dimension of $\mathbf{h}$ through one-hot encoding. In the first case, a typical DL model is trained to predict the attribute of interest, e.g., race, gender, hair color, or background, which the main model should avoid

"using" for its prediction. Note that training a classifier to predict the protected attributes encourages the learning of richer, more diverse latent features associated with them. This approach helps capture subtle, underlying patterns in the data that may otherwise be lost when relying solely on labeled attributes. On the other hand, directly projecting attribute labels through one-hot encoding is easier to implement and computationally less intensive. However, it may not capture the complexity of visual features as effectively as a dedicated bias-capturing classifier.

The sum of the representations, $\mathbf{h} + \mathbf{b}$, is then fed to the final classification layer. Thus, during training, model predictions are computed as $\hat{y}_k^{(j)} = \sigma_k(\mathbf{W}(\mathbf{h}(\mathbf{x}^{(j)}; \boldsymbol{\theta_h}) + \mathbf{b}(\mathbf{x}^{(j)})) + \boldsymbol{\rho})$, where $\mathbf{W}$ and $\boldsymbol{\rho}$ are the parameters of the last linear layer of $h(\cdot)$. By incorporating the biased features $\mathbf{b}$ in the training, we equip the model with the necessary information to consistently account for the bias-aligned samples. This means that the $\mathcal{L}_{\mathcal{B}_\mathcal{A}}$ values are consistently close to 0, preventing the loss spikes, and thus enabling features $\mathbf{h}$ to encode information about the target classes rather than the protected attributes, without having a negative impact on the loss of the bias-aligned samples. In terms of the training process implied by (4), the addition of $\mathbf{b}$ entails invariably large $\sigma_\kappa^{(i)}$ and small $A_0^{(i)}$, thus forcing model updates to depend on the samples of $\mathcal{B}_\mathcal{C}$ consequently eliminating the effect of bias-aligned samples. Having learned a bias-neutral representation $\mathbf{h}$, a final fine-tuning step is required to account for the fact that $\mathbf{b}$ will not be added to input samples at inference time. During this fine-tuning stage, only the final classification layer (i.e., $\mathbf{W}$ and $\boldsymbol{\rho}$) is updated using $\mathbf{h}$ as input. After this step, model predictions are computed using $\hat{y}_k^{(j)} = \sigma_k(\mathbf{W}\mathbf{h}(\mathbf{x}^{(j)}; \boldsymbol{\theta_h}) + \boldsymbol{\rho})$.

While BAdd is found to be very effective in mitigating bias in cases of highly biased datasets, we observe that it does not adversely affect model performance in cases of datasets where bias is much less prevalent (cf. experimental results in the supplementary material). This is an expected behavior because, in low- or no-bias scenarios, the bias-capturing features, $\mathbf{b}$, do not contain information that the model can exploit to predict the target variables. As a result, these features act as noise, which the model naturally learns to ignore without affecting its overall performance.

4. Experimental Setup

4.1. Datasets

Biased-MNIST [4] is an MNIST derivative dataset [20] that serves as a benchmark for bias mitigation methods. It features digits with colored backgrounds, introducing bias through the association of each digit with a specific color. The degree of bias, represented by the probability q of samples belonging to class y and at the same time possessing the attributes t, thus determines the strength of this spurious correlation. We consider four variations of Biased-MNIST with q values of 0.99, 0.995, 0.997, and 0.999, as commonly used in previous works. Biased-CelebA [16] is a subset of the CelebA facial image dataset, which is annotated with 40 binary attributes. Biased-CelebA considers `gender` as the target, while `HeavyMakeup` and `WearingLipstick` serve as the attributes introducing bias. Similarly, Biased-UTKFace [16] is a subset of the facial image UTKFace dataset that is annotated with `gender`, `race`, and `age` labels. `Gender` is the target label, with `race` or `age` considered as protected attributes. In both Biased-CelebA and Biased-UTKFace, the enforced correlation between the target and protected attributes is 0.9. The Corrupted-CIFAR10 dataset [15] consists of 10 classes with texture-related biases uniformly distributed in the training data using four different values of q: 0.95, 0.98, 0.99, and 0.995. Finally, the Waterbirds [28] dataset demonstrates a co-occurrence of 0.95 between waterbirds (or landbirds) and aquatic environments (or terrestrial environments) as background.

Table 1. Fairness of a vanilla `gender` classifier trained on default CelebA w.r.t. potentially biased attributes. Accuracy for the underrepresented groups (e.g., `male-WearingLipstick`) is denoted as "Bias-Conflicting" and the average accuracy across all the subgroups defined by the `gender` and the attribute is denoted as "Unbiased".

Attribute	Accuracy	
	Unbiased	Bias-conflicting
Smiling	98.6	98.5
WearingNecklace	98.1	97.3
WearingEarrings	97.7	96.3
BlondHair	96.9	94.9
Eyeglasses	96.5	94.5
WearingLipstick	95.2	91.1
HeavyMakeup	93.0	86.7

Similar to the Biased-MNIST, we create FB-Biased-MNIST, an extension that enhances the bias introduced by the background color in Biased-MNIST, by injecting foreground color bias into the dataset. Considering the increased complexity of this dataset compared to Biased-MNIST, we opt for lower q values, namely 0.9, 0.95, and 0.99. Furthermore, the UrbanCars dataset is a synthetic

Table 2. CelebA: co-occurrence between `gender` and `WearingLipstick` and `HeavyMakeup` attributes.

Attribute	Co-occurrence	
	Females	Males
WearingLipstick	80.6%	0.06%
HeavyMakeup	66.3%	0.03%

dataset that exhibits a 0.95 co-occurrence between car body type and the background and/or certain objects relevant to urban or rural regions. We also assess the performance of bias mitigation methods on the default CelebA dataset [24] that is devoid of injected biases. To properly select attributes with a measurable degree of bias that could lead to problematic model behavior, we consider the performance disparities of a standard `gender` classifier trained on CelebA with respect to various potentially biased attributes. Subsequently, we identify the top two attributes (`WearingLipstick`, `HeavyMakeup`) with the most significant impact on the model's performance (Tab. 1), as a result of the strong association between these attributes and females (Tab. 2).

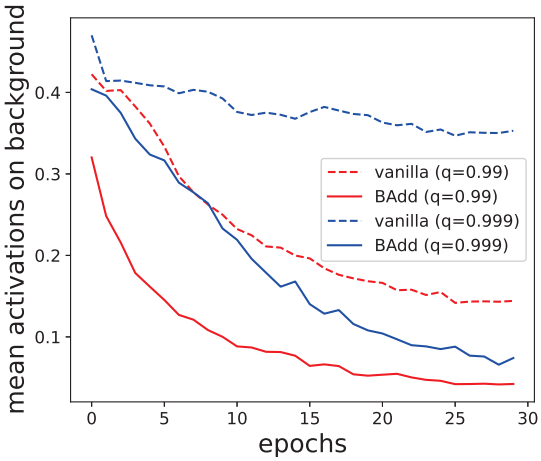
4.2. Evaluation Protocol

Different evaluation setups are used for each dataset, following the conventions of the literature to be comparable with previous works. In particular, following [5, 16, 30], the test sets used for Biased-MNIST and FB-Biased-MNIST are composed using $q = 0.1$ that ensures each digit-color group is equally represented. For Biased-UTKFace and CelebA datasets, we utilize bias-conflicting and unbiased accuracy as in [16, 30]. In particular, bias-conflicting accuracy refers to the accuracy of the under-represented samples (e.g., males wearing lipstick), and unbiased accuracy refers to the average accuracy across all the subgroups defined by the target (i.e., `gender`) and the protected attributes (i.e., `WearingLipstick` and `HeavyMakeup`). The original test set, as shared by the dataset creators, is used in the case of Corrupted-CIFAR10. Regarding the Waterbirds dataset, we employ the average accuracy between different groups and the Worst-Group (WG) accuracy. Finally, for the UrbanCars dataset, we measure the In Distribution Accuracy (I.D. Acc) which is the weighted average accuracy w.r.t. the different groups, where the correlation ratios are the weights. The I.D. Acc is used as a baseline to measure the accuracy drop with respect to the background (BG Gap), co-occurring objects (CoObj Gap), and both the background and co-occurring objects (BG+CoObj Gap). Note that the implementation details are provided in the supplementary material.

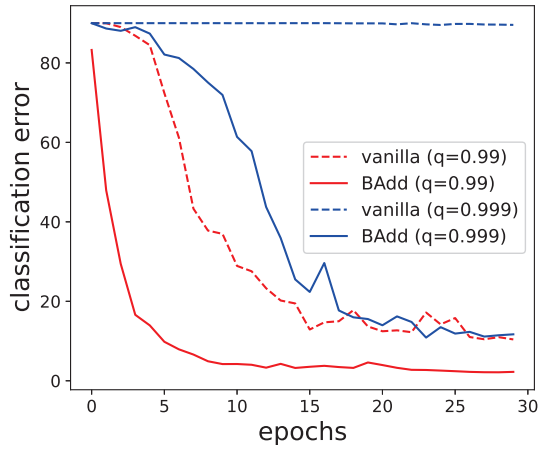
5. Results

5.1. Single Attribute Bias

Table 3 presents the performance of BAdd against nine competing methods. The proposed approach consistently surpasses state-of-the-art, demonstrating accuracy improvements ranging from 0.1% to 0.8% across different q values. Fig. 3 illustrates the mean activations in image regions where bias occurs alongside the corresponding classification errors for both the Vanilla and BAdd approaches. This



(a) Mean activation values of the first convolutional layer on sample backgrounds.



(b) Classification error during the first 30 training epochs.

Figure 3. Vanilla vs BAdd: Mean biased filter activation values and classification error.

makes clear that the proposed method effectively reduces activations in areas where bias appears, leading to significant improvements in classification performance. This is particularly pronounced in experiments with $q = 0.999$, where the vanilla approach struggles with the impact of the biased attribute. Furthermore, the efficacy of BAdd to learn feature representations that are independent of the protected attribute is illustrated in Tab. 4. Specifically, Tab. 4 shows the mean pairwise cosine similarity between 10 variations of each Biased-MNIST test sample, where each variation has a different background color. BAdd leads to similarity values consistently close to 1 for all correlation ratios, which is not the case for the vanilla model that cannot maintain high similarities when the correlation ratio increases (e.g., 0.416 similarity for $q = 0.999$).

Table 3. Evaluation on Biased-MNIST for different bias levels.

Method	q			
	0.99	0.995	0.997	0.999
Vanilla	90.8±0.3	79.5±0.1	62.5±2.9	11.8±0.7
LM [9]	91.5±0.4	80.9±0.9	56.0±4.3	10.5±0.6
Rubi [8]	85.9±0.1	71.8±0.5	49.6±1.5	10.6±0.5
ReBias [4]	88.4±0.6	75.4±1.0	65.8±0.3	26.5±1.4
LfF [25]	95.1±0.1	90.3±1.4	63.7±20.3	15.3±2.9
LNL [17]	86.0±0.2	72.5±0.9	57.2±2.2	18.2±1.2
EnD [39]	94.8±0.3	94.0±0.6	82.7±0.3	59.5±2.3
BC-BB [16]	95.0±0.9	88.2±2.3	82.8±4.2	30.3±11.1
FairKL [5]	97.9±0.0	97.0±0.0	96.2±0.2	90.5±1.5
FLAC [30]	97.9±0.1	96.8±0.0	95.8±0.2	89.4±0.8
BAdd	**98.1**±0.2	**97.3**±0.2	**96.3**±0.2	**91.7**±0.6

Table 4. Mean pairwise cosine similarity between 10 variations of each Biased-MNIST test sample, where each sample variation has a different background color.

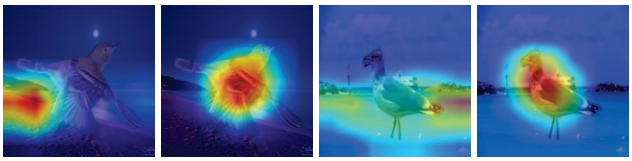
Method	q			
	0.99	0.995	0.997	0.999
Vanilla	0.889	0.854	0.811	0.416
BAdd	0.985	0.985	0.980	0.973

Table 5 illustrates the performance comparison of BAdd against the competing methods on the Biased-UTKFace dataset, where `race` and `age` are considered as protected attributes. Across both protected attributes, the proposed approach outperforms competing methods on bias-conflicting samples, achieving improvements of +1.1% (`race`) and +1.9% (`age`) compared with the second best. In terms of unbiased performance, BAdd exhibits only marginal differences compared to the state-of-the-art methods, with increases of 0.2% (`race`) and decreases of 0.3% (`age`).

Table 5. Evaluation of the proposed method on Biased-UTKFace for two different protected attributes, namely `race` and `age`, with `gender` as the target attribute.

Method	Bias			
	Race		Age	
	Unbiased	Bias-conflicting	Unbiased	Bias-conflicting
Vanilla	87.4±0.3	79.1±0.3	72.3±0.3	46.5±0.2
LNL [17]	87.3±0.3	78.8±0.6	72.9±0.1	47.0±0.1
EnD [39]	88.4±0.3	81.6±0.3	73.2±0.3	47.9±0.6
BC-BB [16]	91.0±0.2	89.2±0.1	79.1±0.3	71.7±0.8
FairKL [5]	85.5±0.7	80.4±1.0	72.7±0.2	48.6±0.6
FLAC [30]	92.0±0.2	92.2±0.7	**80.6**±0.7	71.6±2.6
BAdd	**92.2**±0.2	**93.3**±0.2	80.3±0.8	**73.6**±1.0

In the final single-attribute evaluation scenario, biases stemming from image background or textures are considered. As for the texture biases, the results obtained on the Corrupted-CIFAR10 dataset for four different bias ratios are summarized in Tab. 6. Given the complexity of training a bias-capturing classifier in this scenario, BAdd is implemented using a projection of one-hot vectors representing the texture labels to the feature space of the main model. Notably, BAdd consistently outperforms state-of-the-art across all Corrupted-CIFAR10 variations. Specifically, it achieves improvements of 6.5%, 3.1%, 3.4%, and 1.6% for correlation ratios of 0.95, 0.98, 0.99, and 0.995, respectively.



(a) Vanilla: bias-conflicting (b) BAdd: bias-conflicting (c) Vanilla: bias-aligned (d) BAdd: bias-aligned

Figure 4. Vanilla vs BAdd: GradCam activations on bias-aligned (waterbird with sea background) and bias-conflicting (land bird with sea background) samples of Waterbirds dataset.

Table 7, demonstrates the performance of BAdd on the Waterbirds dataset compared to the state-of-the-art methods for distributionally robust optimization. Here, BAdd reaches the state-of-the-art WG accuracy, i.e., 92.9%, and demonstrates competitive average accuracy, i.e., 93.6%. To further illustrate the effect of BAdd on the behavior of $h(\cdot)$, we visualize GradCam [32] activations for a bias-aligned and a bias-conflicting sample of Waterbirds in Fig. 4. As can be easily noticed, the model trained with BAdd effectively focuses on birds, remaining unaffected by the presence of biases (i.e., background). In contrast, the vanilla model relies primarily on the background for its predictions.

Table 6. Evaluation on Corrupted-CIFAR10.

Method	q			
	0.95	0.98	0.99	0.995
Vanilla	39.4±0.6	30.1±0.7	25.8±0.3	23.1±1.2
EnD [39]	36.6±4.0	34.1±4.8	23.1±1.1	19.4±1.4
ReBias [4]	43.4±0.4	31.7±0.4	25.7±0.2	22.3±0.4
LfF [25]	50.3±1.6	39.9±0.3	33.1±0.8	28.6±1.3
FairKL [5]	50.7±0.9	41.5±0.4	36.5±0.4	33.3±0.4
FLAC [30]	53.0±0.7	46.0±0.2	39.3±0.4	34.1±0.5
BAdd	**59.5**±0.5	**49.1**±0.3	**42.7**±0.2	**35.7**±0.6

Table 7. Evaluation on Waterbirds.

Method	WG Acc.	Avg. Acc.
JTT [23]	86.7±1.5	93.3±0.3
DISC [43]	88.7±0.4	93.8±0.7
GroupDro [28]	90.6±1.1	91.8±0.3
DFR [19]	**92.9**±**0.2**	**94.2**±**0.4**
BAdd	**92.9**±**0.3**	93.6±0.2

Table 8. Evaluation on FB-Biased-MNIST.

Method	q		
	0.9	0.95	0.99
Vanilla	82.5±0.8	57.9±1.7	25.5±0.6
EnD [39]	82.5±1.0	57.5±2.0	25.7±0.8
BC-BB [16]	80.9±2.4	66.0±2.4	40.9±3.4
FairKL [5]	87.6±0.8	61.6±2.6	42.0±1.1
FLAC [30]	84.4±0.8	63.1±1.7	32.4±1.1
BAdd	**95.6**±**0.3**	**89.0**±**1.8**	**69.5**±**2.5**

5.2. Multi-Attribute Bias

As previously discussed, evaluating bias mitigation performance solely in single-attribute scenarios provides an initial assessment but fails to capture the complexities of real-world settings. In this section, we present the performance of BAdd in two multi-attribute bias evaluation setups, namely on FB-Biased-MNIST and CelebA datasets.

As depicted in Tab. 8, competing methods struggle to effectively mitigate bias on the FB-Biased-MNIST dataset, while BAdd consistently outperforms the second-best performing methods by significant margins of 8%, 23%, and 27.5% for q of 0.9, 0.95, and 0.99, respectively. Notably, even in an artificial dataset like FB-Biased-MNIST, existing approaches struggle to address multiple biases. Table 9 demonstrates the performance of BAdd on UrbanCars, a dataset with artificially injected bias that is much more challenging than FB-Biased-MNIST. As observed, most methods struggle to address both the background and the co-occurring object biases. The only exception is LLE, which employs architectural modifications and specific bias-oriented augmentations to tackle each type of bias. However, LLE requires extensive pre-processing, including object segmentation, making its application to other CV datasets very effort-intensive or even infeasible. Finally, as an example of a real-world dataset without artificially injected biases, we use the default CelebA dataset, where `gender` is the target attribute and multiple biases are present. As shown in Tab. 10, BAdd consistently improves performance for the attributes introducing bias, achieving absolute accuracy improvements of +3.5% and +5.5% for the bias-conflicting samples and +1.1% and +2.1% average accuracy across the subgroups compared to the second-best performing methods.

Table 9. Evaluation on UrbanCars.

Method	I.D. Acc	BG Gap	CoObj Gap	BG+CoObj Gap
LfF [25]	97.2	-11.6	-18.4	-63.2
JTT [23]	95.9	-8.1	-13.3	-40.1
Debian [21]	98.0	-14.9	-10.5	-69.0
GroupDro [28]	91.6	-10.9	-3.6	-16.4
DFR [19]	89.7	-10.7	-6.9	-45.2
LLE [22]	96.7	**-2.1**	-2.7	-5.9
BAdd	91.0±0.7	-4.3±0.4	**-1.6**±**1.0**	**-3.9**±**0.4**

Table 10. Evaluation of the proposed method on CelebA for multiple attributes introducing bias, namely `WearingLipstick` and `HeavyMakeup`. `Gender` is the target attribute.

Method	Biases			
	WearingLipstick		HeavyMakeup	
	Unbiased	Bias-conflicting	Unbiased	Bias-conflicting
Vanilla	95.2±0.3	91.1±0.6	93.0±0.8	86.7±1.6
EnD [39]	95.1±0.4	91.0±0.7	92.3±0.7	85.3±1.5
BC-BB [16]	91.6±2.6	85.8±5.1	89.7±2.3	81.8±4.5
FairKL [5]	82.7±0.4	74.7±0.3	84.4±0.9	77.9±1.2
FLAC [30]	95.4±0.3	91.6±0.5	93.2±0.3	87.2±0.7
BAdd	**96.5**±**0.2**	**95.1**±**0.4**	**95.3**±**0.5**	**92.7**±**1.1**

6. Conclusion

In this work, we propose a method for bias mitigation in CV DL models, termed BAdd. The proposed method injects bias-capturing features in the features of a model to force the model parameter updates to rely only on unbiased samples, thus leading to bias-neutral representations. The main requirement for BAdd is to either have access to the labels introducing bias within the dataset or to be able to train attribute label predictors on another dataset where these labels are available. Also, note that existing bias identification methods [18, 46] can be employed to infer such labels. However, the exploration of bias identification techniques falls outside the scope of this work. Through a comprehensive experimental evaluation, we show that the proposed approach surpasses the state-of-the-art in single-attribute and more profoundly in multi-attribute bias scenarios.

Acknowledgments

This research was supported by the EU Horizon Europe projects MAMMOth (grant no. 101070285), ELIAS (grant no. 101120237), and ELLIOT (grant no. 101214398).

References

[1] Tameem Adel, Isabel Valera, Zoubin Ghahramani, and Adrian Weller. One-network adversarial fairness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2412–2420, 2019. 3

[2] Sumyeong Ahn, Seongyoon Kim, and Se-Young Yun. Mitigating dataset bias by using per-sample gradient. *arXiv preprint arXiv:2205.15704*, 2022. 2

[3] Mohsan Alvi, Andrew Zisserman, and Christoffer Nellåker. Turning a blind eye: Explicit removal of biases and variation from deep neural network embeddings. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018. 3

[4] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *International Conference on Machine Learning*, pages 528–539. PMLR, 2020. 2, 3, 5, 7, 1

[5] Carlo Alberto Barbano, Benoit Dufumier, Enzo Tartaglione, Marco Grangetto, and Pietro Gori. Unbiased supervised contrastive learning. *arXiv preprint arXiv:2211.05568*, 2022. 2, 3, 6, 7, 8

[6] Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning: Limitations and Opportunities*. fairmlbook.org, 2019. http://www.fairmlbook.org. 1

[7] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. Recommender systems survey. *Knowledge-based systems*, 46:109–132, 2013. 1

[8] Remi Cadene, Corentin Dancette, Matthieu Cord, Devi Parikh, et al. Rubi: Reducing unimodal biases for visual question answering. *Advances in neural information processing systems*, 32, 2019. 2, 3, 7

[9] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4069–4082, 2019. 2, 3, 7

[10] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65, 2018. 1

[11] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. 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, 2019. 1

[12] Simone Fabbrizzi, Symeon Papadopoulos, Eirini Ntoutsi, and Ioannis Kompatsiaris. A survey on bias in visual datasets. *Computer Vision and Image Understanding*, 223: 103552, 2022. 1

[13] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6202–6211, 2019. 1

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[15] Dan Hendrycks and Thomas G Dietterich. Benchmarking neural network robustness to common corruptions and surface variations. *arXiv preprint arXiv:1807.01697*, 2018. 2, 5

[16] Youngkyu Hong and Eunho Yang. Unbiased classification through bias-contrastive and bias-balanced learning. *Advances in Neural Information Processing Systems*, 34: 26449–26461, 2021. 2, 3, 5, 6, 7, 8, 1

[17] Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9012–9020, 2019. 2, 3, 7

[18] Younghyun Kim, Sangwoo Mo, Minkyu Kim, Kyungmin Lee, Jaeho Lee, and Jinwoo Shin. Discovering and mitigating visual biases through keyword explanation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11082–11092, 2024. 8

[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. 8

[20] Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998. 5

[21] Zhiheng Li, Anthony Hoogs, and Chenliang Xu. Discover and mitigate unknown biases with debiasing alternate networks. In *European Conference on Computer Vision*, pages 270–288. Springer, 2022. 3, 8

[22] Zhiheng Li, Ivan Evtimov, Albert Gordo, Caner Hazirbas, Tal Hassner, Cristian Canton Ferrer, Chenliang Xu, and Mark Ibrahim. A whac-a-mole dilemma: Shortcuts come in multiples where mitigating one amplifies others. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20071–20082, 2023. 1, 2, 3, 8

[23] 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. PMLR, 2021. 3, 8

[24] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015. 2, 6

[25] 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. 3, 7, 8

[26] Shikai Qiu, Andres Potapczynski, Pavel Izmailov, and Andrew Gordon Wilson. Simple and fast group robustness by automatic feature reweighting. In *International Conference on Machine Learning*, pages 28448–28467. PMLR, 2023. 3

[27] Ryan Ramos, Vladan Stojnic, Giorgos Kordopatis-Zilos, Yuta Nakashima, Giorgos Tolias, and Noa Garcia. Processing and acquisition traces in visual encoders: What does clip know about your camera? In *International Conference on Computer Vision*, 2025. 2

[28] 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. 1, 2, 3, 5, 8

[29] Ioannis Sarridis, Christos Koutlis, Symeon Papadopoulos, and Christos Diou. Vb-mitigator: An open-source framework for evaluating and advancing visual bias mitigation. *arXiv preprint arXiv:2507.18348*. 2

[30] Ioannis Sarridis, Christos Koutlis, Symeon Papadopoulos, and Christos Diou. Flac: Fairness-aware representation learning by suppressing attribute-class associations. *arXiv preprint arXiv:2304.14252*, 2023. 2, 3, 6, 7, 8, 1

[31] Ioannis Sarridis, Christos Koutlis, Symeon Papadopoulos, and Christos Diou. Towards fair face verification: An in-depth analysis of demographic biases. *arXiv preprint arXiv:2307.10011*, 2023. 1

[32] 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. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 7

[33] Robik Shrestha, Kushal Kafle, and Christopher Kanan. An investigation of critical issues in bias mitigation techniques. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1943–1954, 2022. 2

[34] Robik Shrestha, Kushal Kafle, and Christopher Kanan. Occamnets: Mitigating dataset bias by favoring simpler hypotheses. In *European Conference on Computer Vision*, pages 702–721. Springer, 2022. 2

[35] Tomáš Sixta, Julio Jacques Junior, Pau Buch-Cardona, Eduard Vazquez, and Sergio Escalera. Fairface challenge at eccv 2020: Analyzing bias in face recognition. In *European conference on computer vision*, pages 463–481. Springer, 2020. 1

[36] Jiaming Song, Pratyusha Kalluri, Aditya Grover, Shengjia Zhao, and Stefano Ermon. Learning controllable fair representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2164–2173. PMLR, 2019. 3

[37] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014. 1

[38] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020. 1

[39] Enzo Tartaglione, Carlo Alberto Barbano, and Marco Grangetto. End: Entangling and disentangling deep representations for bias correction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13508–13517, 2021. 2, 3, 7, 8

[40] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 2

[41] Tianlu Wang, Jieyu Zhao, Mark Yatskar, Kai-Wei Chang, and Vicente Ordonez. Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5310–5319, 2019. 2, 3

[42] Zeyu Wang, Klint Qinami, Ioannis Christos Karakozis, Kyle Genova, Prem Nair, Kenji Hata, and Olga Russakovsky. Towards fairness in visual recognition: Effective strategies for bias mitigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8919–8928, 2020. 3

[43] 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. 3, 8

[44] Qizhe Xie, Zihang Dai, Yulun Du, Eduard Hovy, and Graham Neubig. Controllable invariance through adversarial feature learning. *Advances in neural information processing systems*, 30, 2017. 3

[45] Quanshi Zhang, Wenguan Wang, and Song-Chun Zhu. Examining cnn representations with respect to dataset bias. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018. 2, 3

[46] Zaiying Zhao, Soichiro Kumano, and Toshihiko Yamasaki. Language-guided detection and mitigation of unknown dataset bias. *arXiv preprint arXiv:2406.02889*, 2024. 8

[47] Zhang Zhifei, Song Yang, and Qi Hairong. Age progression/regression by conditional adversarial autoencoder. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017. 2

[48] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017. 2