Fair Attribute Classification through Latent Space De-biasing

Vikram V. Ramaswamy, Sunnie S. Y. Kim, Olga Russakovsky
Princeton University
{vr23, suhk, olgarus}@cs.princeton.edu

Abstract

How does this affect training data with generating based off of similar face? Or do GANs take care of this?

Fairness in visual recognition is becoming a prominent and critical topic of discussion as recognition systems are deployed at scale in the real world. Models trained from data in which target labels are correlated with protected attributes (e.g., gender, race) are known to learn and exploit those correlations. In this work, we introduce a method for training accurate target classifiers while mitigating biases that stem from these correlations. We use GANs to generate realistic-looking images, and perturb these images in the underlying latent space to generate training data that is balanced for each protected attribute. We augment the original dataset with this generated data, and empirically demonstrate that target classifiers trained on the augmented dataset exhibit a number of both quantitative and qualitative benefits. We conduct a thorough evaluation across multiple target labels and protected attributes in the CelebA dataset, and provide an in-depth analysis and comparison to existing literature in the space. Code can be found at https://github.com/princetonvisualai/gan-debiasing.

Builds on dataset to make more fair.

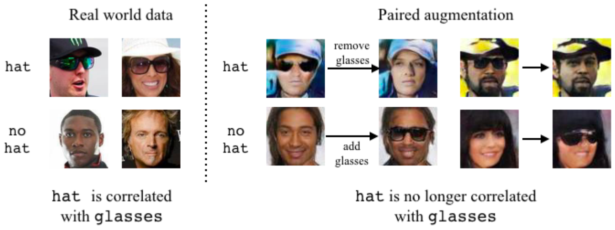


Figure 1: Training a visual classifier for an attribute (e.g., `hat`) can be complicated by correlations in the training data. For example, the presence of hats can be correlated with the presence of glasses. We propose a dataset augmentation strategy using Generative Adversarial Networks (GANs) that successfully removes this correlation by adding or removing glasses from existing images, creating a balanced dataset.

1. Introduction

Large-scale supervised learning has been the driving force behind advances in visual recognition. Recently, however, there has been a growing number of concerns about the disparate impact of these visual recognition systems. Face recognition systems trained from datasets with an under-representation of certain racial groups have exhibited lower accuracy for those groups [9]. Activity recognition models trained on datasets with high correlations between the activity and the gender expression of the depicted person have over-amplified those correlations [46]. Computer vision systems are statistical models that are trained to maximize accuracy on the majority of examples, and they do so by exploiting the most discriminative cues in a dataset, potentially learning spurious correlations. In this work, we introduce a new framework for training computer vision models that aims to mitigate such concerns, illustrated in Figure 1.

One proposed path for building 'fairer' computer vision systems is through a 'fairer' data collection process. Works such as [9, 43] propose techniques for better sampling data to more accurately represent all people. Creating a perfectly balanced dataset, however, is infeasible in many cases. With the advances in Generative Adversarial Networks (GANs) [17], several works propose using generated data to augment real-world datasets [12, 35, 42]. These methods have been growing in computational and algorithmic complexity (e.g., [35, 42] adding multiple loss functions to GAN training), necessitating access to a sufficient number of inter-sectional real-world samples. In contrast, we demonstrate a simple and novel data augmentation technique that uses a single GAN trained on a biased real-world dataset.

" Fairer " Data collection
- Doesn't solve our problem

Illustrative example: Consider our example from Figure 1. Our goal is to train a visual recognition model that recognizes the presence of an attribute, such as wearing a hat. Suppose in the real world wearing a hat is correlated with wearing glasses—for example, because people often wear both hats and sunglasses outside and take them off inside. This correlation may be reflected in the training data, and a classifier trained to recognize a hat may rely on the presence of glasses. Consequently, the classifier may fail to recognize a hat in the absence of glasses, and vice versa.

We propose using a GAN to generate more images with hats but not glasses and images with glasses but not hats, such that `WearingHat` is de-correlated from `Glasses` in the training data, by making perturbations in the latent space. Building on work by Denton et al. [14], which demonstrates

Using a GAN to de correlate data

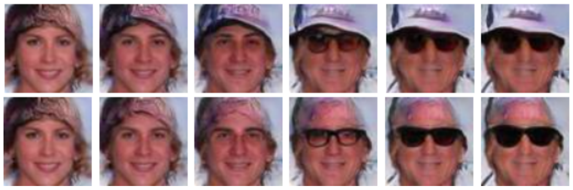


Figure 2: Consider a GAN trained on a biased real-world dataset of faces where the presence of hats is correlated with the presence of glasses. Naively moving in a direction that adds glasses also adds a hat (*Top*). We learn a direction in the latent space that allows us to add glasses, while not adding a hat (*Bottom*). Note that attributes apart from the target attribute can change.

a method for learning interpretable image manipulation directions, we propose an improved latent vector perturbation method that allows us to preserve the `WearingHat` attribute while changing the `Glasses` attribute (Figure 2).

Protected attributes: Our goal is to examine and mitigate biases of sensitive attributes such as gender expression, race, or age in visual classifiers. However, visual manipulations or explicit classifications along these dimensions have the potential to perpetuate harmful stereotypes (see [23]). Hence in our illustrations, we use `Glasses` as the protected attribute, as it has a clear visual signal. In the quantitative experimental results, we report our findings on the more sensitive protected attributes of gender expression and age.

Contributions: We propose a method for perturbing vectors in the GAN latent space that successfully de-correlates target and protected attributes and allows for generating a de-biased dataset, which we use to augment the real-world dataset. Attribute classifiers trained with the augmented dataset achieve quantitative improvements in several fairness metrics over both baselines and prior work [35, 36, 41], while maintaining comparable average precision. Furthermore, we analyze the CelebA [28] attributes with respect to label characteristics[1], discriminability, and skew, and discuss how these factors influence our method's performance. We also evaluate our design choices with ablation studies and the results demonstrate the effectiveness of our augmentation method.[2]

2. Related Work

De-biasing models: The effect of gender and racial bias on AI models has been well documented [8, 9, 22, 40, 41]. Models trained on biased data sometimes even amplify the existing biases [46]. Tools such as AI Fairness 360 [6] and REVISE [38] surface such biases in large-scale datasets and enable preemptive analysis. In parallel, various work propose methods for mitigating unwanted dataset biases from influencing the model. Oversampling techniques [7, 15] duplicate minority samples in imbalanced data to give them higher weight in training. Some work propose to mitigate bias through adversarial learning [40, 45] or through learning separate classifiers for each protected attribute [33, 41]. Other work improve fairness by introducing constraints [29] or regularization terms [3] during training. Contrary to these algorithmic approaches, our work aims to mitigate biases by training the model with a generated de-biased dataset.

previous mitigation strategies

Generating and perturbing images using GANs: Generative Adversarial Network (GAN) [17] is a popular class of generative models composed of a generator and a discriminator trained in an adversarial setting. Over the past few years, a number of works [18, 24, 25, 27, 34] improved GANs to generate more realistic images with better stability. Shen et al. [37] show that the latent space of GANs have semantic meaning and demonstrate facial attributes editing through latent space manipulation. Denton et al. [14] propose a method to evaluate how sensitive a trained classifier is to such image manipulations, and find several attributes that affect a smiling classifier trained on CelebA. Balakrishnan et al. [4] use GANs to generate synthetic images that differ along specific attributes while preserving other attributes, and use them to measure algorithmic bias of face analysis algorithms. Unlike [4, 14] who use the GAN-generated images to evaluate models, our work uses these generated images to train better attribute classification models.

GAN handles this

Data Snooping?

Using GANs to augment datasets: Several works use GANs to augment datasets for low-shot [20] and long-tail [48] recognition tasks, whereas our work focuses specifically on de-biasing classifiers affected by dataset bias. More related to our work are [12, 35, 36] which leverage GANs to generate less biased data. Choi et al. [12], given access to a small, unlabeled, and unbiased dataset, detect bias in a large and potentially biased dataset, and learn a generator that generates unbiased data at test time. Sattigeri et al. [35] train a GAN with a modified loss function to achieve demographic parity or equality of odds in the generated dataset. Sharmanska et al. [36] use an image-to-image translation GAN to generate more minority samples and create a balanced dataset. While [12, 35, 36] require training a new GAN for each bias they want to correct, our method uses a single GAN trained on a biased dataset to augment all attributes.

3. Method

We study a class of problems where a protected attribute is correlated with a target label in the data $\mathcal{X}$, influencing target label prediction. Let t be the target label (e.g., `WearingHat` in the running example from Figure 1) and g be the protected attribute (e.g., gender expression or `Glasses` from our running example) with $t, g \in \{-1, 1\}$. To mitigate the effect of unwanted dataset bias, we aim to generate a balanced set of synthetic images $\mathcal{X}_{syn}$ where the protected attribute and target label are de-correlated.

t - target label

g - protected Attribute

$t, g \in \{-1, 1\}$

[1] We observe several discrepancies in the CelebA [28] attribute labels and categorize the attributes into three categories: inconsistently labeled, gender-dependent, and gender-independent.

[2] Code for all our experiments can be found at https://github.com/princetonvisualai/gan-debiasing.

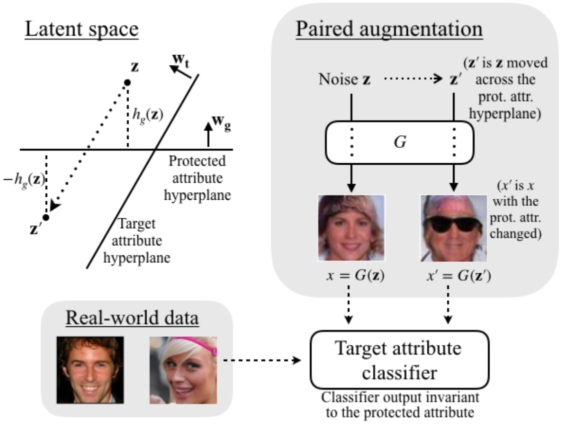

Latent space
$\mathbf{z}$
$\mathbf{w_t}$
$h_g(\mathbf{z})$
$\mathbf{w_g}$
Protected attribute hyperplane
$-h_g(\mathbf{z})$
$\mathbf{z}'$
Target attribute hyperplane
Paired augmentation
Noise $\mathbf{z}$
$\mathbf{z}'$
($\mathbf{z}'$ is $\mathbf{z}$ moved across the prot. attr. hyperplane)
G
(x' is x with the prot. attr. changed)
$x = G(\mathbf{z})$
$x' = G(\mathbf{z}')$
Real-world data
Target attribute classifier
Classifier output invariant to the protected attribute

Figure 3: *(Top left)* Our latent vector perturbation method. For each $\mathbf{z}$ sampled from the latent space of a trained GAN, we compute $\mathbf{z}'$ such that its target attribute score remains the same (according to $\mathbf{w_t}$) while its protected attribute score is negated (according to $\mathbf{w_g}$). *(Top right)* We add images $G(\mathbf{z})$ and $G(\mathbf{z}')$ to our training set, and train a target attribute classifier on both the real-world data and the generated de-biased data.

Concretely, let f_t be a function from images to binary labels that approximates the target label t, and f_g be a function from images to binary labels that approximates the protected attribute g. We learn these classifiers in a supervised fashion with the original data.[3] We now want to generate synthetic data $\mathcal{X}_{syn}$ with the property that for $\mathbf{x} \in \mathcal{X}_{syn}$:

$$P\left[f_t(\mathbf{x}) = 1 | f_g(\mathbf{x}) = 1\right] = P\left[f_t(\mathbf{x}) = 1\right], \tag{1}$$

such that attributes t and g are de-correlated.

De-biased dataset creation: To create $\mathcal{X}_{syn}$, we use a GAN trained on real images $\mathcal{X}$ whose generator G generates a synthetic image $\mathbf{x}$ from a random latent vector $\mathbf{z} \in \mathcal{Z}$. We can assign semantic attribute labels to these images using the learned functions $f_t(\mathbf{x})$ and $f_g(\mathbf{x})$. However, as the GAN inherits correlations from its training data, a random sampling of $\mathbf{z}$ will produce an $\mathcal{X}_{syn}$ with similar correlations and biases as $\mathcal{X}$. Hence, we propose a latent vector perturbation method that allows us to generate a de-biased $\mathcal{X}_{syn}$.

We sample a random set of latent vectors $Z \subset \mathcal{Z}$ (inheriting the biases) and train classifiers $h_t, h_g : \mathcal{Z} \to [-1, 1]$ in the latent space that approximate $f_t \circ G$ and $f_g \circ G$, respectively. That is, we train classifiers h_t with input $\mathbf{z}$ and output $f_t(G(\mathbf{z}))$, and h_g with input $\mathbf{z}$ and output $f_g(G(\mathbf{z}))$.

Given a vector $\mathbf{z}$, we generate a complementary vector $\mathbf{z}'$ with the same (predicted) target label but the opposite (predicted) protected attribute label, or

$$h_t(\mathbf{z}') = h_t(\mathbf{z}), \quad h_g(\mathbf{z}') = -h_g(\mathbf{z}). \tag{2}$$

We note that this data generation method is agnostic to the type of classifier used to compute h.

In our work, we assume that the latent spaces is approximately linearly separable in the semantic attributes, as observed and empirically validated by Denton et al. [14]. In this case, h_t and h_g can be represented as linear models (hyperplanes) $\mathbf{w_t}$ and $\mathbf{w_g}$ with intercepts b_t and b_g for the target and protected attributes respectively. We can derive a closed-form solution for $\mathbf{z}'$ as[4]

$$\mathbf{z}' = \mathbf{z} - 2\left(\frac{\mathbf{w_g}^T\mathbf{z} + b_g}{1 - (\mathbf{w_g}^T\mathbf{w_t})^2}\right)\left(\mathbf{w_g} - (\mathbf{w_g}^T\mathbf{w_t})\mathbf{w_t}\right). \tag{3}$$

This latent vector perturbation method is illustrated in Figure 3 (*Top left*). A similar idea of hyperplane projection was presented in Zhang et al. [45], although for a different goal of adversarial training. The sampling process results in a complementary image pair:

- $\mathbf{x} = G(\mathbf{z})$ with target label $f_t(G(\mathbf{z}))$ and protected attribute label $f_g(G(\mathbf{z}))$
- $\mathbf{x}' = G(\mathbf{z}')$ with target label $f_t(G(\mathbf{z}))$ and protected attribute label $-f_g(G(\mathbf{z}))$,

creating de-biased data $\mathcal{X}_{syn}$. We train our target attribute classifier with $\mathcal{X}$ and $\mathcal{X}_{syn}$, as shown in Figure 3.

We label the generated images $\mathbf{x}$ and $\mathbf{x}'$ both with $f_t(\mathbf{x})$ because it allows us to capture the target attribute labels better than using $f_t(\mathbf{x})$ and $f_t(\mathbf{x}')$. It is likely that the accuracy of f_t is higher for the overrepresented group, and $\mathbf{x}$ will more often belong to the overrepresented group and $\mathbf{x}'$ to the underrepresented group. However, other design choices are possible in our approach—for example, we could use $h_t(\mathbf{z})$ and $h_t(\mathbf{z}')$ instead (after thresholding appropriately) or only use $\mathbf{z}$ for which $f_t(\mathbf{x}) = f_t(\mathbf{x}')$. We compare these different design choices experimentally in Section 4.2.

Advantages: Our data augmentation method has several attractive properties:

1. We use a single GAN trained on the biased real-world dataset to augment multiple target labels and protected attributes. This is in contrast to prior works like [35, 12] that require training a GAN for every pair of target and protected attributes.
2. By augmenting samples $\mathbf{z}$ generated from (approximately) the original data distribution the GAN was trained on and maintaining their target attribute scores, our method preserves the intra-class variation of the images.
3. The samples $\mathbf{z}$ and $\mathbf{z}'$ are generated to simulate the independence goal of Equation 1. By construction, $\mathbf{z}'$ maintains $\mathbf{z}$'s target label $f_t(G(\mathbf{z}))$ and takes on the opposite protected attribute label $-f_g(G(\mathbf{z}))$.
4. Our method generalizes to multiple protected attributes g. We demonstrate how our method can simultaneously augment two protected attributes in Section 4.3 when we compare our work to Sharmanska et al. [36].

[3] f_t is equivalent to the baseline classifier in Section 4.1.

[4] Derivations are in the appendix (Section A). $\|\mathbf{w_t}\| = \|\mathbf{w_g}\| = 1$.

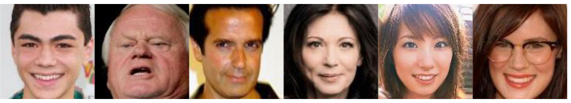


Figure 4: Examples of CelebA `StraightHair` labels. Some of these are labeled as having `StraightHair` (1st, 3rd, 5th) and some as not (2nd, 4th, 6th). We deemed this attribute as *inconsistently labeled*.

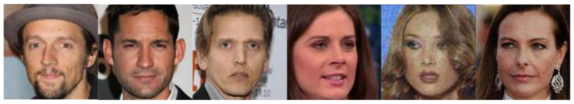


Figure 5: Examples of CelebA `Young` labels. The first three images are labeled `Male`, `Young` while the last three images are labeled `not Male`, `not Young`, even though the first three appear older than the last three. We deemed this attribute as *gender-dependent*.

4. Experiments

In this section, we study the effectiveness of our data augmentation method on training fairer attribute classifiers. We first describe our experiment setup and compare our results to those of a baseline classifier. We then discuss how different factors influence our method's performance, and finally compare our work to several prior works.

Dataset and attributes categorization: Given the task of training attribute classifiers that are not dependent on gender expression, we require a dataset that has target labels, as well as gender expression labels. CelebA [28] is a dataset with 2,022,599 images of celebrity faces, each with 40 binary attributes labels. We assume the `Male` attribute corresponds to gender expression.[5] Among the other 39 attributes, we use 26 of them that have between 1% and 99% fraction of positive images for each gender expression.[6] However, we noticed several discrepancies among the attribute labels, and decided to categorize the attributes into three categories: *inconsistently labeled*, *gender-dependent*, and *gender-independent*.

We categorized attributes as *inconsistently labeled* when we visually examined sets of examples and found that we often disagreed with the labeling and could not distinguish between positive and negative examples. This category includes `StraightHair` shown in Figure 4, as well as `BigLips`, `BigNose`, `OvalFace`, `PaleSkin`, and `WavyHair`.[7] While we report results on these attributes for completeness in Section 4.1, classifiers trained on these attributes may behave erratically.

Of the remaining attributes with more consistent labeling, we found that some attribute labels are *gender-dependent*. That is, images are labeled to have (or not have) these attributes based on the perceived gender. For example in Figure 5, we observe that the images labeled as `Young` and `Male` appear much older than the images labeled as `Young` and `not Male`. Other attributes in this category are `ArchedBrows`, `Attractive`, `BushyBrows`, `PointyNose` and `RecedingHair`.

The *gender-independent* attribute labels appear to be reasonably consistent among annotators, and do not appear to depend on the gender expression. We classified 14 attributes into this category: `Bangs`, `BlackHair`, `BlondHair`, `BrownHair`, `Chubby`, `Earrings`, `EyeBags`, `Glasses`, `GrayHair`, `HighCheeks`, `MouthOpen`, `NarrowEyes`, `Smiling`, and `WearingHat`. While we use the label 'gender-independent' we note that these attributes can still be correlated with gender expression—for example `Earrings` are much more common among images labeled as `not Male` than those labeled as `Male`.

Implementation details: To generate images, we use a Progressive GAN [24] with a 512-D latent space trained on the CelebA [28] training set from the PyTorch GAN Zoo [16]. We use 10,000 synthetic images, labeled with baseline attribute classifiers, and learn hyperplanes (h_t, h_g) in the latent space with scikit-learn's [31] linear SVM implementation.

For all attribute classifiers, we use ResNet-50 [21] pretrained on ImageNet [32] as the base architecture. We replace the linear layer in ResNet with two linear layers with the hidden layer of size 2,048. Dropout and ReLU are applied between these. The inputs are 64×64 images and their target attribute labels. We train all models with the binary cross entropy loss for 20 epochs with a batch size of 32. We use the Adam [26] optimizer with a learning rate of 1e-4. We save the model with the smallest loss on a validation set that has the same distribution as the training set.

The baseline model is trained on the CelebA training set $\mathcal{X}$ with 162,770 images. Our model is trained on $\mathcal{X}$ and the balanced synthetic dataset $\mathcal{X}_{syn}$ (160,000 pairs of images).[8] Results are reported on the CelebA test set unless noted otherwise. Error bars are 95% confidence intervals estimated through bootstrapping. We note that we use a single GAN to construct the de-biased dataset for each target attribute, and then train separate classifiers for each target attribute. We also emphasize that protected attribute labels are only used in learning h_g and in evaluation.

Evaluation Metrics: We use *average precision (AP)* to

[5] Consistent with the dataset annotation and with the literature, we adopt the convention of using `Male` as our protected attribute. It is not clear if this label denotes assigned sex at birth, gender identity, or gender expression (socially perceived gender). Since the images were labeled by a professional labeling company [28], we assume that the annotation refers to the perceived gender, or gender expression. Moreover, this attribute is annotated in a binary fashion. We would like to point out that none of these attributes (assigned sex at birth, gender identity, nor gender expression) are binary, however, we use these labels as is for our goal of de-biasing classifiers.

[6] We don't use `Blurry` as it has very few positive images (≈ 5%). We don't use `WearingNecklace` as the cropped images used in the GAN from [16] don't display the neck.

[7] We note that for `BigNose`, we found that while there were some images that were easy to classify as having a big nose, or not having a big nose, most images were between these two extremes, and we believe that different annotators marked these 'in-between' images differently. The same is true for the attribute `BigLips`.

[8] We trained classifiers using different number of synthetic pairs for 4 different attributes, and found that AP stabilizes after 160,000 pairs, which is what we used to train our classifiers.

measure the accuracy of the classifiers. AP is a threshold-invariant accuracy metric that summarizes the precision and recall curve. We use this metric to ensure that our models learn a reasonable classification rule. AP, however, does not capture a classifier's behavior on different protected classes, and in fact, we expect to see a slight dip in overall AP when our model improves on some of the fairness metrics.

Multiple metrics have been proposed to measure fairness of a model [19, 44, 46, 10, 11] and each of these measures a different notion of fairness. In our work, we use three metrics for comprehensive understanding. First, we measure the *difference in equality of opportunity (DEO)*, i.e. the absolute difference between the false negative rates for both gender expression, as in Lokhande et al. [29][9].

As our second fairness metric, we use the *bias amplification (BA)* metric proposed by Wang and Russakovsky [39]. Intuitively, BA measures how much more often a target attribute is predicted with a protected attribute than the ground truth value. Let $P_{t|g}$ be the fraction of images with protected attribute g that have target attribute t, $P_{\hat{t}|g}$ be the fraction of images with protected attribute g that are predicted to have target attribute t, $P_{t,g}$ be the fraction of images with target t and protected attribute g, and P_t and P_g be the fraction of images with attribute t and g respectively. For each pair of target and protected attribute values, we add $(P_{t|g} - P_{\hat{t}|g})$ if $P_{t,g} > P_t P_g$ and $-(P_{t|g} - P_{\hat{t}|g})$ otherwise. A negative value implies that bias now exists in a different direction than in the training data.

Both DEO and BA fluctuate based on the chosen classification threshold. Hence, as our final fairness metric, we use a threshold-invariant metric that measures the *divergence between score distributions (KL)* [11] defined as follows: Suppose $s_{g,t}$ represents a smoothed histogram of classifier scores of a certain protected attribute label and a target label, appropriately normalized as a probability distribution of the scores. For each target attribute label t, we measure $KL\big[s_{g=-1,t} \| s_{g=1,t}\big] + KL\big[s_{g=1,t} \| s_{g=-1,t}\big]$. That is, we measure the divergence of $g{=}-1$ and $g{=}1$ score distributions, separately for positive and negative attribute samples. This is a stricter notion of *equalized odds*[19].

4.1. Comparison with the baseline

To start, we compare our model (i.e. target classifiers trained using both the balanced synthetic datasets $\mathcal{X}_{syn}$ and the real dataset $\mathcal{X}$) with a baseline model trained using just $\mathcal{X}$. In Table 1, we show results on the four metrics, averaged for each of the three attribute categories. As expected, our model performs better on all three fairness metrics, DEO, BA and KL, while maintaining comparable AP. For gender-independent attributes, AP drops from 83.9 to 83.0, while DEO improves from 16.7 to 13.9, BA improves from 0.3 to 0.0 and KL improves from 1.1 to 0.9. For gender-dependent attributes, the fairness metrics improve over the baseline, but the improvements are smaller compared to those of gender-independent attributes. Later in Section 5, we demonstrate an extension of our augmentation method with an improved performance on the gender-dependent attributes.

Attr. type	AP ↑		DEO ↓	
	Baseline	Ours	Baseline	Ours
Incons.	**66.3 ± 1.8**	65.2 ± 1.9	21.5 ± 4.4	**16.5 ± 4.2**
G-dep	**78.6 ± 1.4**	77.8 ± 1.4	25.7 ± 3.5	**23.4 ± 3.6**
G-indep.	**83.9 ± 1.5**	83.0 ± 1.6	16.7 ± 5.0	**13.9 ± 5.2**
Attr. type	BA ↓		KL ↓	
	Baseline	Ours	Baseline	Ours
Incons.	2.1 ± 0.6	**0.5 ± 0.6**	1.7 ± 0.3	**1.3 ± 0.4**
G-dep	2.3 ± 0.5	**1.6 ± 0.5**	1.3 ± 0.2	**1.2 ± 0.2**
G-indep.	0.3 ± 0.6	**0.0 ± 0.5**	1.1 ± 0.5	**0.9 ± 0.6**

Table 1: Comparison of our model (i.e. attribute classifier trained with our data augmentation method) to the baseline model. Arrows indicate which direction is better. Numbers are averages over all attributes within the specific category. As expected, we have slightly lower AP than the baseline, but perform better on the three fairness metrics, DEO, BA, and KL.

Additionally, we conduct score change evaluations suggested by Denton et al. [14] and measure the change in target attribute score as we perturb the protected attribute in images. Specifically, we measure the classifier score difference between $G(\mathbf{z})$ and $G(\mathbf{z}')$. This evaluation helps understand how the protected attribute influences a trained classifier's output. We find that the model trained with our augmentation method consistently has a smaller change in score than the baseline: 0.09 vs. 0.12 for inconsistently labeled, 0.07 vs. 0.11 for gender-dependent, and 0.06 vs. 0.09 for gender-independent attributes. We also observe that the baseline score changes are higher when we try to construct underrepresented samples. Consider the attribute `ArchedBrows` where only 2.3% of the training set images are labeled to have `ArchedBrows`, and appear masculine. When we construct a $\mathbf{z}'$ with this target and protected value, the baseline classifier's score changes by 0.41. On the other hand, when we try to construct an image that is without `ArchedBrows` and appears feminine, which comprises 33.7% of the training set, the baseline classifier score only changes by 0.094. This could be due to the errors that the baseline classifier makes on underrepresented images during synthetic image labeling, or could imply that underrepresented attributes are harder to maintain during image manipulations.

We next examine several factors that could influence our method, including how easy the protected attribute is to learn compared to the target attribute and how data skew affects our method. We discuss the former here and provide more information about the latter in the appendix (Section C).

Discriminability of attributes: Nam et al. [30] recently ob-

[9] In our experiments, we choose a calibrated threshold on the validation set, i.e, a threshold that ensures that we make the same number of positive predictions as the ground truth, to compute both DEO and BA. We tried other ways of choosing the threshold, such as choosing the one that gives the best F_1 score on a validation set, and while the values varied, they did not change our findings.

Protected Attribute	Improvement over baseline ↑					
	DEO		BA		KL	
	Easy	Hard	Easy	Hard	Easy	Hard
`Glasses` (0,19)	–	**4.1**	–	**0.9**	–	**0.0**
`Gender` (2, 17)	0.8	**3.2**	0.0	**0.4**	-0.2	**0.2**
`Young` (15, 4)	-0.2	**2.1**	0.2	**1.0**	-0.2	**0.0**

Table 2: Improvement over baseline for different fairness metrics when using different protected attributes. Next to the protected attribute are numbers of attributes that are 'easier' and 'harder' to learn, compared to the protected attribute. Columns 'Easy' ('Hard') show the averages of all non-inconsistent target attributes that are easier (harder) for a classifier to learn. We note that our method works better when the target attribute is 'harder' to learn.

served that correlations among attributes affect a classifier only if the protected attribute is 'easier' to learn than the target attribute. Inspired by their observation, we conduct a two-step experiment to understand how the relative discriminability of attributes affects our method's effectiveness.

First, we put a pair of CelebA attributes in competition to assess their relative discriminability. Experiment details are in the appendix. We find that gender expression is one of the easiest attributes to learn (`Gender` is easier than all but `Glasses` and `WearingHat`), which may be why gender bias is prevalent in many models. On the other hand, `Young` is relatively hard for a model to learn (`Young` is harder to learn than all but 4 other attributes), so its correlation with other attributes may not be as influential.

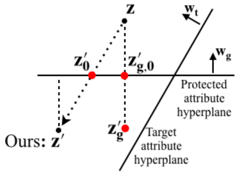
Next, to understand how the relative discriminability of attributes affects our method's performance, we train target attribute classifiers for gender-dependent and gender-independent attributes, using `Young` and `Glasses` as protected attributes. In Table 2, we report our method's improvement over baseline in the three fairness metrics. For each protected attribute, we report the average improvement separately for 'easier' and 'harder' target attributes. While training with our augmentation method generally outperforms the baseline on the three fairness metrics, as expected, the improvement is greater for target attributes that are harder to learn than the protected attribute, for example, for `Young`, the improvement in DEO over baseline is -0.2 for easy target attributes, and 2.1 for hard target attributes.

Skew of the dataset: The *skew* of a target attribute t is measured following the literature [41] as $\frac{\max(P_{-1}, P_1)}{P_{-1}+P_1}$ where P_{-1} is the number of images with $t{=}1$ and protected attribute label $g{=}-1$, and P_1 is the number of images with $t{=}1$ and protected attribute label $g{=}1$. We find that our augmentation method is most effective on attributes with low to moderate skew. Full details are in the appendix.

4.2. Ablation studies

We now examine the design choices made in our method.

Removal of $\mathbf{z}'$ samples: First, we evaluate the effect of $G(\mathbf{z}')$ on the classifier. We train a classifier with just $G(\mathbf{z})$ and the real dataset $\mathcal{X}$, and compare its performance against the performance of our model, trained with $G(\mathbf{z})$, $G(\mathbf{z}')$, and $\mathcal{X}$ on the gender-dependent and gender-independent attributes. While the new classifier's AP is higher than that of our model (82.9 vs. 82.6), all fairness metrics are worse: DEO is higher (19.7 vs. 16.1), BA is higher (1.1 vs. 0.5) and KL is higher (1.6 vs 1.3). All numbers were calculated on the validation set. In fact, it performs worse on the fairness metrics than the baseline model trained on $\mathcal{X}$. This result suggests that simply synthesizing more images with a GAN and adding them to the training data does not improve the model but rather hurts performance. Possible reasons include the image and label noise of $G(\mathbf{z})$ and the skew of $G(\mathbf{z})$ being worse than the original data the GAN was trained on. The fairness metrics improve only when we add $G(\mathbf{z}')$, and make the training data more balanced.



Perturbation	AP ↑	
	G-dep	G-indep
$\mathbf{z}'_{g,0}$	74.0	79.9
$\mathbf{z}'_g$	69.6	77.3
$\mathbf{z}'_0$	74.4	79.8
$\mathbf{z}'$ (ours)	**76.0**	**81.4**

Figure 6: Comparison of different perturbation choices. We train attribute classifiers using only synthetic images generated from the perturbations, and measure the mean AP over all target attributes on the validation set. The classifier trained with $\mathbf{z}'$ (our choice) has the highest AP.

Choice of $\mathbf{z}'$: Next, we evaluate our choice of $\mathbf{z}'$ through examining a number of alternative perturbation choices visualized in Figure 6. We train classifiers on just the generated data for gender-dependent and gender-independent attributes and compare the overall AP on the validation set. As expected, training with $\mathbf{z}'$ (our choice) has the highest AP.

Filtering z's and using different labels for synthetic images: Since we hallucinate labels for the synthetic images, some of these labels may be incorrect and harm our classifier. We try three different ways of addressing this issue: First, we try learning hyperplanes with different fractions of positive and negative samples. We find that while this improves the hyperplane accuracy, the downstream classifiers trained with samples generated using different hyperplanes have similar performances. For the second and third methods, we use the original hyperplanes learned in our method, but vary the vectors/labelling used. We remove points that are incorrectly classified by the baseline classifier after perturbing the latent vector from $\mathbf{z}$ to $\mathbf{z}'$, i.e, we remove all points wherein $f_t(G(\mathbf{z})) \neq f_t(G(\mathbf{z}'))$, and use the remaining synthetic images and the real dataset to train the classifiers. Third, we label the synthetic images $G(\mathbf{z})$ and $G(\mathbf{z}')$ with $h_t(\mathbf{z})$, and use these labels to train the classifiers. We compare their performance to our method on the validation set. We find that these two methods result in a slight drop in AP (79.8 when using h_t scores, 82.1 when removing incorrectly classified points, and 82.6 for our method), as well as a small drop in the fairness metrics (the average DEO is 18.1 when using h_t scores, 17.4 when removing incorrectly classified

	Fairness GAN [35]				Ours	
	Dem. Par.		Eq. Opp.		(Synthetic only)	
Gender exp. g	$g{=}{-1}$	$g{=}1$	$g{=}{-1}$	$g{=}1$	$g{=}{-1}$	$g{=}1$
FPR ↓	0.52	0.26	0.42	**0.17**	**0.22**	0.39
FNR ↓	0.18	0.41	0.21	0.44	**0.06**	**0.27**
Error ↓	0.30	0.28	0.29	0.23	**0.21**	**0.18**
Error Rate ↓	0.22		0.29		**0.20**	

Table 3: Comparison of the `Attractive` classifier trained using synthetic data from Fairness GAN [35] and the classifier trained using our pair-augmented synthetic data. The latter (ours) outperforms on most metrics.

points, and 16.1 for our method), suggesting that our current labeling of the synthetic images works well. Full results are in the appendix (Section D).

4.3. Comparison with prior work

In this section, we compare our method to few recent works [35, 36, 41]. One of the current challenges in the space of AI fairness is the lack of standardized benchmarks and metrics. While some of this stems from the complexity of the problem at hand (where it is difficult and even counter-productive to use a single fairness definition), in the computer vision community, we believe that more effort should be made to provide thorough comparison between methods. Each work we consider here uses slightly different evaluation protocols and benchmarks. We made comparisons to the best of our ability, and hope that our work helps enable more standardization and empirical comparisons.

Fairness GAN: Sattigeri et al. [35] use GANs to create datasets that achieve either demographic parity (Dem. Par.) or equality of opportunity (Eq. Opp.). They train classifiers for the `Attractive` attribute on just the generated data, using gender expression as the protected attribute. We train classifiers with our pair-augmented synthetic data to mimic the conditions of Fairness GAN, and evaluate both on the CelebA test data. Comparison results are in Table 3. Our model performs better on most metrics, even though we use a single GAN to augment all attributes.

Contrastive examples generated by image-to-image translation GANs: Sharmanska et al. [36] propose a different method for balancing a biased dataset using StarGAN [13], a class of image-to-image translation GANs. They use two protected attributes, age and gender expression, and create a balanced dataset by creating contrastive examples, i.e. images of different ages and gender, for each image in the training set. They train a `Smiling` classifier with the augmented dataset, and propose making a prediction at test time only when the classifier makes the same prediction on the image and their contrastive examples. We extend our method to incorporate multiple protected attributes, and use gradient descent to find three points $\{z_i'\}_{i\in\{1,2,3\}}$ in the latent space that preserve the target attribute score and flip either the gender expression score, the age score, or both. This process gives us three synthetic images per training image, with which we train a `Smiling` classifier. To ensure that the error rates are similar across all four protected groups—(`Young`, `Male`), (`Young`, `not Male`), (`not Young`, `Male`), (`not Young`, `not Male`)—they measure the the mean difference in the false positive and false negative rates between all pairs of protected groups. We reproduce their method to ensure that the results are reported on the same test set. We find that our model performs better in terms of the mean difference in FNR (0.34 versus their 0.54) and FPR (0.23 compared to their 0.46).

Skew	Method	AP ↑	DEO ↓	BA ↓	KL ↓
Low/ Mod.	Dom. Ind.	**83.4** ± **1.3**	7.0 ± 3.1	**-0.1** ± **0.5**	0.8 ± 0.7
	Ours	81.4 ± 1.5	**6.0** ± **3.0**	**-0.1** ± **0.5**	**0.3** ± **0.1**
High	Dom. Ind.	**80.7** ± **1.6**	**14.9** ± **5.6**	**-0.4** ± **0.5**	**0.8** ± **1.0**
	Ours	80.4 ± 1.5	23.9 ± 5.5	0.9 ± 0.4	1.5 ± 0.6

Table 4: Comparison of our method with domain independent training [41]. Numbers reported are the mean over all gender-dependent and gender-independent attributes on the test set. We note that we perform better than domain-independent training for attributes with low to moderate skew.

Method	AP ↑	DEO ↓	BA ↓	KL ↓
Weighted	79.6 ± 1.6	**5.7** ± **4.2**	**-2.8** ± **0.5**	**0.5** ± **0.4**
Adversarial	81.3 ± 1.6	23.9 ± 4.4	1.5 ± 0.5	0.6 ± 0.5
Ours	**81.5** ± **1.5**	16.7 ± 4.7	0.5 ± 0.5	1.0 ± 0.5

Table 5: Comparison of our method with weighted and adversarial training from [41]. Numbers reported are the mean over all gender-dependent and gender-independent attributes on the test set. We note that the weighted model overall performs better on the fairness metrics, however, the large negative BA suggests that the model now has bias in the opposite direction, to the extent that the AP drops. The adversarial model performs significantly worse than ours on DEO and BA, and marginally better on KL.

Effective training strategies for bias mitigation: Wang et al. [41] quantitatively compare different techniques for bias mitigation, including weighted training [7, 15], adversarial training with losses inspired by [2, 45], and their proposed *domain discriminative* and *domain independent* training. We compare our method to their best performing domain independent training method where they learn separate classifiers for each protected attribute class and combine them to leverage any shared information. We report results for all gender-dependent and gender-independent attributes in Table 4. We find that our method performs better for attributes with low to moderate skew (<0.7)—DEO is 6.0 compared to 7.0, KL is 0.3 compared to 0.8—whereas domain independent training performs better for attributes with high skew—DEO is 23.9 compared to 14.9, KL is 1.5 compared to 0.8. This result is consistent with our earlier observation that our method works well for low to moderately skewed datasets. Wang et al also use a simpler weighted training method that reweights samples such that the protected attribute classes have equal weight and an adversarial training method that uses a minimax objective to maximize the classifier's accuracy on the objective while minimizing an adversary's ability to predict the protected attribute from the learned features. For weighted and adversarial training methods, we report results in Table 5. We find that while the weighted model overall performs well on the fairness metrics, it has a strongly negative BA (-2.7 versus our 0.5) indicating that bias is now in

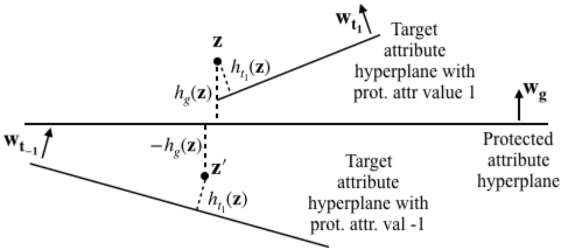


Figure 7: Computing $\mathbf{z}'$ when the target attribute hyperplanes for each protected attribute class are very different.

the opposite direction, and a low AP (79.6 versus our 81.5) suggesting that it makes incorrect predictions to reduce bias. For adversarial training, our method does better overall, with lower DEO (16.7 versus 23.9) and lower BA (0.5 versus 1.5).

5. Extensions of our method

In this final section, we study two natural extensions of our method: using domain-dependent hyperplanes in place of the current domain-independent hyperplanes, and directly augmenting a real image dataset with GAN-inversion.

Domain-dependent hyperplanes: Our method implicitly assumes the learned hyperplane $\mathbf{w_t}$ behaves equally well for all $\mathbf{z}$, irrespective of the value of $f_g(G(\mathbf{z}))$. However, for gender-dependent attributes, the hyperplane learned using samples with $f_g(G(\mathbf{z}))$=1 may be very different from that learned using samples with $f_g(G(\mathbf{z}))$=-1.

For these attributes, we extend our method to learn per-domain target attribute hyperplanes: $\mathbf{w}_{t_1}, b_{t_1}$ for points with $f_g(G(\mathbf{z}))$=1 and $\mathbf{w}_{t_{-1}}, b_{t_{-1}}$ for points with $f_g(G(\mathbf{z}))$=-1. For $\mathbf{z}$ with $f_g(G(\mathbf{z}))$=1, we find $\mathbf{z}'$ such that

$$\begin{aligned} \mathbf{w}_{t_{-1}}^T(\mathbf{z}') + b_{t_{-1}} &= \mathbf{w}_{t_1}^T(\mathbf{z}) + b_{t_1}, \text{ and} \\ \mathbf{w_g}^T\mathbf{z}' + b_g &= -\mathbf{w_g}^T(\mathbf{z}) - b_g \end{aligned} \tag{4}$$

as shown in Figure 7. In order to compute $\mathbf{z}'$ that satisfies the above constraints, while minimizing $||\mathbf{z}-\mathbf{z}'||_2$, we note that all constraints are linear, hence the feasible region is the intersection of several hyperplanes. Starting from a point in this region, in each iteration, we find a new location of the point using gradient descent, then project it back onto the feasible region to maintain the constraints.

If $\mathbf{w}_{t_1}$ and $\mathbf{w}_{t_{-1}}$ are similar, these constraints are the same as Equation 2 and this method of computing $\mathbf{z}'$ collapses to the first. We compare results of training a classifier that is augmented with images computed with domain-independent hyperplanes and with that using images computed with domain-dependent hyperplanes for all gender-dependent and gender-independent attributes over the validation set. We find that for gender-dependent attributes, using domain-dependent hyperplanes improves the fairness metrics considerably (DEO reduces from 21.4 to 17.2, BA reduces from 1.5 to 0.4, KL reduces from 1.2 to 1.0), without losing accuracy. However, for gender-independent attributes, we do not see significant improvement, suggesting that $\mathbf{w_t}$ is similar to both $\mathbf{w}_{t_1}$ and $\mathbf{w}_{t_{-1}}$. Full results are in Table 6.

Attr. type	AP ↑		DEO ↓	
	Dom-ind	Dom-dep	Dom-indep	Dom-dep
G-dep	**78.1 ± 1.5**	**78.1 ± 1.4**	21.4 ± 4.0	**17.2 ± 4.0**
G-indep	84.5 ± 1.5	**84.6 ± 1.6**	13.9 ± 4.3	**13.1 ± 4.6**
Attr. type	BA ↓		KL ↓	
	Dom-indep	Dom-dep	Dom-indep	Dom-dep
G-dep	1.5 ± 0.5	**0.4 ± 0.5**	1.2 ± 0.2	**1.0 ± 0.3**
G-indep	**0.1 ± 0.4**	0.2 ± 0.4	**0.9 ± 0.5**	**0.9 ± 0.6**

Table 6: Comparison of classifiers that use domain-dependent hyperplanes vs. domain-independent hyperplanes to compute z'. We see a significant improvement among Gender-dependent attributes when we use Domain-dependent hyperplanes. Numbers are reported on the validation set.

	AP ↑	DEO ↓	BA ↓	KL ↓
Without	**82.6 ± 1.5**	1.5 ± 2.3	**1.3 ± 0.4**	**1.0 ± 0.5**
With inv.	82.4 ± 1.5	**1.4 ± 2.3**	**1.3 ± 0.4**	**1.0 ± 0.5**

Table 7: Comparison of our classifiers (without) to classifiers trained using data augmented with a GAN-inversion module (with inv.). Numbers reported are the mean over all gender-dependent and gender-independent attributes on the validation set. We do not see an appreciable improvement.

Augmenting real images with GAN-inversion: Our method operates in the GAN latent space and can only augment images that are generated from latent vectors, and so, only the GAN-generated images. Recently, several GAN-inversion methods have been proposed [1, 5, 47]. These methods invert a real image $\mathbf{x}_{real} \in \mathcal{X}$ to a vector $\mathbf{z}_{inv}$ in the latent space of a trained GAN. Using Zhu et al. [47], we tried directly augmenting the original dataset by perturbing $\mathbf{z}_{inv}$ to $\mathbf{z}'_{inv}$ with our method, creating $\mathbf{x}'_{real}$=$G(\mathbf{z}'_{inv})$ with the same target label and the opposite protected label of $\mathbf{x}_{real}$. When we trained classifiers with datasets augmented in this way, however, we did not see an appreciable improvement, despite the more complex procedure (Table 7).

6. Conclusions

We introduced a GAN-based data augmentation method for training fairer attribute classifiers when correlations between the target label and the protected attribute (such as gender expression) might skew the results. We report results across a large number of attributes and metrics, including comparisons with existing techniques. We also analyze in detail when our method is the most effective. Our findings show the promise of augmenting data in the GAN latent space in a variety of settings. We hope our detailed analyses and publicly available code serve as a stepping stone for future explorations in this very important space.

Acknowledgements: This work is supported by the National Science Foundation under Grant No. 1763642 and the Princeton First Year Fellowship to SK. We also thank Arvind Narayanan, Deniz Oktay, Angelina Wang, Zeyu Wang, Felix Yu, Sharon Zhang, as well as the Bias in AI reading group for helpful comments and suggestions.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2StyleGAN: How to embed images into the StyleGAN latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 8

[2] 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)*, 2018. 7

[3] Sina Baharlouei, Maher Nouiehed, Ahmad Beirami, and Meisam Razaviyayn. Rényi fair inference. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2020. 2

[4] Guha Balakrishnan, Yuanjun Xiong, Wei Xia, and Pietro Perona. Towards causal benchmarking of bias in face analysis algorithms. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2020. 2

[5] David Bau, Jun-Yan Zhu, Jonas Wulff, William Peebles, Hendrik Strobelt, Bolei Zhou, and Antonio Torralba. Seeing what a GAN cannot generate. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 8

[6] Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias, Oct. 2018. 2

[7] Steffen Bickel, Michael Brückner, and Tobias Scheffer. Discriminative learning under covariate shift. *Journal of Machine Learning Research*, 10(Sep):2137–2155, 2009. 2, 7

[8] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357, 2016. 2

[9] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 77–91, 2018. 1, 2

[10] Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. Building classifiers with independency constraints. In *2009 IEEE International Conference on Data Mining Workshops*, pages 13–18. IEEE, 2009. 5

[11] Mingliang Chen and Min Wu. Towards threshold invariant fair classification. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI)*, 2020. 5

[12] Kristy Choi, Aditya Grover, Rui Shu, and Stefano Ermon. Fair generative modeling via weak supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2020. 1, 2, 3, 14

[13] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 7

[14] Emily Denton, Ben Hutchinson, Margaret Mitchell, and Timnit Gebru. Image counterfactual sensitivity analysis for detecting unintended bias. In *CVPR 2019 Workshop on Fairness Accountability Transparency and Ethics in Computer Vision*, 2019. 1, 2, 3, 5

[15] Charles Elkan. The foundations of cost-sensitive learning. In *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI)*, volume 17, pages 973–978. Lawrence Erlbaum Associates Ltd, 2001. 2, 7

[16] FAIR HDGAN. Pytorch GAN Zoo. 4

[17] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014. 1, 2

[18] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of Wasserstein GANs. In *Advances in Neural Information Processing Systems*, pages 5767–5777, 2017. 2

[19] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems*, pages 3315–3323, 2016. 5

[20] Bharath Hariharan and Ross Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3018–3027, 2017. 2

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 4

[22] Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *Proceedings of European Conference on Computer Vision (ECCV)*, pages 793–811. Springer, 2018. 2

[23] Khari Johnson. Google Cloud AI removes gender labels from Cloud Vision API to avoid bias, 02 2020. 2

[24] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018. 2, 4

[25] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4401–4410, 2019. 2

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015. 4

[27] Steven Liu, Tongzhou Wang, David Bau, Jun-Yan Zhu, and Antonio Torralba. Diverse image generation via self-conditioned GANs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[28] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3730–3738, 2015. 2, 4

[29] Vishnu Suresh Lokhande, Aditya Kumar Akash, Sathya N. Ravi, and Vikas Singh. FairALM: Augmented lagrangian method for training fair models with little regret. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2020. 2, 5

[30] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: Training debiased classifier from biased classifier. In *Advances in Neural Information Processing Systems*, 2020. 5, 12

[31] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. 4

[32] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 4

[33] Hee Jung Ryu, Hartwig Adam, and Margaret Mitchell. InclusiveFaceNet: Improving face attribute detection with race and gender diversity. In *International Conference on Machine Learning (ICML) FATML Workshop*, 2018. 2

[34] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016. 2

[35] Prasanna Sattigeri, Samuel C Hoffman, Vijil Chenthamarakshan, and Kush R Varshney. Fairness GAN: Generating datasets with fairness properties using a generative adversarial network. *IBM Journal of Research and Development*, 63(4/5):3–1, 2019. 1, 2, 3, 7

[36] Viktoriia Sharmanska, Lisa Anne Hendricks, Trevor Darrell, and Novi Quadrianto. Contrastive examples for addressing the tyranny of the majority, 2020. 2, 3, 7

[37] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of GANs for semantic face editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9243–9252, 2020. 2

[38] Angelina Wang, Arvind Narayanan, and Olga Russakovsky. REVISE: A tool for measuring and mitigating bias in visual datasets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2

[39] Angelina Wang and Olga Russakovsky. Directional bias amplification. *arXiv preprint arXiv:2102.12594*, 2021. 5

[40] 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 (ICCV)*, pages 5310–5319, 2019. 2

[41] Zeyu Wang, Klint Qinami, Ioannis 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 (CVPR)*, 2020. 2, 6, 7

[42] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. FairGAN: Fairness-aware generative adversarial networks. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 570–575. IEEE, 2018. 1

[43] Kaiyu Yang, Klint Qinami, Li Fei-Fei, Jia Deng, and Olga Russakovsky. Towards fairer datasets: Filtering and balancing the distribution of the people subtree in the imagenet hierarchy. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 547–558, 2020. 1

[44] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the International Conference on World Wide Web (WWW)*, pages 1171–1180, 2017. 5

[45] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES)*, pages 335–340, 2018. 2, 3, 7

[46] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2017. 1, 2, 5

[47] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. Indomain GAN inversion for real image editing. In *Proceedings of European Conference on Computer Vision (ECCV)*, 2020. 8

[48] Xinyue Zhu, Yifan Liu, Jiahong Li, Tao Wan, and Zengchang Qin. Emotion classification with data augmentation using generative adversarial networks. In Dinh Phung, Vincent S. Tseng, Geoffrey I. Webb, Bao Ho, Mohadeseh Ganji, and Lida Rashidi, editors, *Advances in Knowledge Discovery and Data Mining (KDD)*, pages 349–360, Cham, 2018. Springer International Publishing. 2

Appendix

In this supplementary document, we provide additional details on certain sections of the main paper.

Section A: We derive a closed form solution for $\mathbf{z}'$ which allows us to easily manipulate latent vectors in the latent space (Section 3).

Section B We provide attribute-level results and further analysis of our main experiments (Section 4.1).

Section C: We discuss some factors that influence (or not) our method's effectiveness.

Section D: We provide more details on the ablation studies (Section 4.2).

Section E: We investigate how many images with protected attribute labels our method requires to achieve the desired performance.

A. Derivation

In Section 3 of the main paper, we describe a method to compute perturbations within the latent vector space, such that the protected attribute score changes, while the target attribute score remains the same. More formally, if h_t is a function that approximates the target attribute score, and h_g is a function that approximates the protected attribute score, for every latent vector $\mathbf{z}$, we want to compute $\mathbf{z}'$ such that

$$h_t(\mathbf{z}') = h_t(\mathbf{z}), \quad h_g(\mathbf{z}') = -h_g(\mathbf{z}). \tag{5}$$

We assume that the latent space $\mathcal{Z}$ is approximately linearly separable in the semantic attributes. h_t and h_g thus can be represented as linear models $\mathbf{w_t}$ and $\mathbf{w_g}$, normalized as $||\mathbf{w_t}|| = 1, ||\mathbf{w_g}|| = 1$, for the target and protected attribute respectively, with intercepts b_t and b_g.

Equation 5 thus reduces to

$$\mathbf{w_t}^T\mathbf{z} + b_t = \mathbf{w_t}^T\mathbf{z}' + b_t, \quad \mathbf{w_g}^T\mathbf{z}' + b_g = -\mathbf{w_g}^T\mathbf{z} - b_g. \tag{6}$$

Simplifying, we get

$$\mathbf{w_t}^T(\mathbf{z}' - \mathbf{z}) = 0, \quad \mathbf{w_g}^T(\mathbf{z}' + \mathbf{z}) + 2b_g = 0. \tag{7}$$

These equations have infinitely many solutions, we choose the solution that minimizes the distance between $\mathbf{z}$ and $\mathbf{z}'$. This is true if $\mathbf{z}' - \mathbf{z}$ is in the span of $\{\mathbf{w_g}, \mathbf{w_t}\}$. Hence, we can represent $\mathbf{z}' - \mathbf{z} = \alpha\mathbf{w_t} + \beta\mathbf{w_g}$, and we get:

$$\mathbf{w_t}^T(\mathbf{z}' - \mathbf{z}) = 0 \tag{8}$$

$$\mathbf{w_t}^T(\alpha\mathbf{w_t} + \beta\mathbf{w_g}) = 0 \tag{9}$$

$$\Rightarrow \alpha = -\beta\mathbf{w_t}^T\mathbf{w_g} \tag{10}$$

$$\mathbf{w_g}^T((\mathbf{z}' - \mathbf{z}) + 2\mathbf{z}) + 2b_g = 0 \tag{11}$$

$$\mathbf{w_g}^T(\alpha\mathbf{w_t} + \beta\mathbf{w_g} + 2\mathbf{z}) + 2b_g = 0 \tag{12}$$

$$-\beta(\mathbf{w_t}^T\mathbf{w_g})^2 + \beta + 2\mathbf{w_g}^T\mathbf{z} + 2b_g = 0 \tag{13}$$

$$\Rightarrow (1 - (\mathbf{w_t}^T\mathbf{w_g})^2)\beta = -2(\mathbf{w_g}^T\mathbf{z} + b_g) \tag{14}$$

$$\Rightarrow \beta = -2\frac{(\mathbf{w_g}^T\mathbf{z} + b_g)}{(1 - (\mathbf{w_t}^T\mathbf{w_g})^2)} \tag{15}$$

$$\Rightarrow \alpha = 2\frac{(\mathbf{w_g}^T\mathbf{z} + b_g)(\mathbf{w_t}^T\mathbf{w_g})}{(1 - (\mathbf{w_t}^T\mathbf{w_g})^2)} \tag{16}$$

This gives us a closed form solution for $\mathbf{z}'$:

$$\mathbf{z}' = \mathbf{z} - 2\left(\frac{\mathbf{w_g}^T\mathbf{z} + b_g}{1 - (\mathbf{w_g}^T\mathbf{w_t})^2}\right)\left(\mathbf{w_g} - (\mathbf{w_g}^T\mathbf{w_t})\mathbf{w_t}\right). \tag{17}$$

As a quick verification, we confirm that this value of $\mathbf{z}'$ maintains changes the protected attribute score, and maintains the target attribute score:

$$\begin{aligned}
&h_g(\mathbf{z}') \\
&= \mathbf{w_g}^T\mathbf{z}' + b_g \\
&= \mathbf{w_g}^T\left[\mathbf{z} - 2\left(\frac{\mathbf{w_g}^T\mathbf{z} + b_g}{1 - (\mathbf{w_g}^T\mathbf{w_t})^2}\right)\left(\mathbf{w_g} - (\mathbf{w_g}^T\mathbf{w_t})\mathbf{w_t}\right)\right] + b_g \\
&= \mathbf{w_g}^T\mathbf{z} - 2\left(\frac{\mathbf{w_g}^T\mathbf{z} + b_g}{1 - (\mathbf{w_g}^T\mathbf{w_t})^2}\right)\left(1 - (\mathbf{w_g}^T\mathbf{w_t})\mathbf{w_g}^T\mathbf{w_t}\right) + b_g \\
&= \mathbf{w_g}^T\mathbf{z} - 2(\mathbf{w_g}^T\mathbf{z} + b_g) + b_g = -\mathbf{w_g}^T\mathbf{z} - b_g = -h_g(\mathbf{z})
\end{aligned}$$

$$\begin{aligned}
&h_a(\mathbf{z}') \\
&= \mathbf{w_t}^T\mathbf{z}' + b_t \\
&= \mathbf{w_t}^T\left[\mathbf{z} - 2\left(\frac{\mathbf{w_g}^T\mathbf{z} + b_g}{1 - (\mathbf{w_g}^T\mathbf{w_t})^2}\right)\left(\mathbf{w_g} - (\mathbf{w_g}^T\mathbf{w_t})\mathbf{w_t}\right)\right] + b_t \\
&= \mathbf{w_t}^T\mathbf{z} - 2\left(\frac{\mathbf{w_g}^T\mathbf{z}}{1 - (\mathbf{w_g}^T\mathbf{w_t})^2}\right)\left(\mathbf{w_t}^T\mathbf{w_g} - (\mathbf{w_g}^T\mathbf{w_t})\right) + b_t \\
&= \mathbf{w_t}^T\mathbf{z} + b_t = h_t(\mathbf{z})
\end{aligned}$$

B. Attribute-level results

We provide attribute-level results and further analysis of our main experiments (Section 4.1 of the main paper).

B.1 Linear separability of latent space

Our paired augmentation method assumes that the latent space is approximately linearly separable in the semantic attributes. Here we investigate to what extent this assumption holds for different attributes. As described in the main paper, the attribute hyperplanes were estimated with 10,000 samples using linear SVM.

In Table 8, we report hyperplane accuracy and AP, measured on 160,000 synthetic samples, as well as the percentage of positive samples and the skew of the CelebA training set. The skew is calculated as $\frac{\max(N_{g=-1,a=1}, N_{g=1,a=1})}{N_{g=-1,a=1}+N_{g=1,a=1}}$ where $N_{g=-1,a=1}$ is the number of samples with protected attribute label $g{=}-1$ (perceived as not male) and target label 1 (positive) and $N_{g=1,a=1}$ defined likewise. The protected attribute class with more positive samples is noted in the skew column. We observe that most attributes are well separated with the estimated hyperplanes, except for those with high skew that have too few examples from underrepresented subgroups.

For completeness, we also report our model's improvement over the baseline model on the four evaluation metrics. We did not find immediate correlations between the hyperplane quality with the downstream model performance.

B.2 Changes in baseline score

We next evaluate how well we are able to maintain the target attribute score when perturbing the latent vector. We use the change in the baseline classifier as a proxy to measure the target attribute score. We note that this measurement is flawed because the baseline classifier is known to perform worse on minority examples, however, we believe that this measurement still leads to some valuable insights. For each attribute, we measure the the absolute change in baseline score $|f_t(G(\mathbf{z}) - f_t(G(\mathbf{z}'))|$ over 5000 images, and compute averages based on what we expect the target and protected attribute values of $G(\mathbf{z}')$ to be. We plot this versus the fraction of images in the real world dataset that have these target and protected values (Figure 8). We find that there is a strong negative correlation. This could be because the target attribute is harder to maintain in this case, or because the baseline classifier has a tendency to misclassify minority samples.

Another question that we were interested in was interactions between different attributes as we create balanced synthetic datasets for different attributes. We measured the change in baseline classifier score for different targets t' when trying to maintain target attribute t and found that some attributes changed drastically when creating a balanced dataset for any attribute (Table 9). For example, the attribute Attractive changed by a large amount irrespective of which target attribute we were trying to preserve. This suggests that some of these attributes are more sensitive to latent space manipulations.

C. Factors of influence

In this section, we discuss in more detail how some factors influence (or not) our method's effectiveness (Section 4.1 of the main paper).

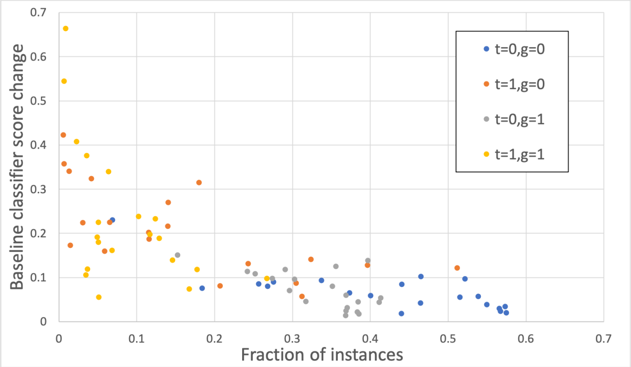


Figure 8: We plot average absolute change in the baseline classifier score versus the fraction of images in the dataset that have the corresponding ground truth labels. We separate them based on what the new ground truth values should be, for each attribute. We find that the score change is larger when creating an image with minority labels. This could be because we are unable to maintain the target attribute in this case or because the baseline classifier performs worse on minority images.

C.1 Skew of attributes

For some attributes, the majority of the positive samples come from one gender expression. For example, ArchedBrows has a skew of 0.92 towards $g{=}-1$, that is, 92% of positive ArchedBrows samples have gender expression label $g{=}-1$. To understand the effect of data skew on our method's performance, we ran experiments with differently skewed data. From the 162,770 CelebA training set images, we created slightly smaller training sets where the attribute of interest (e.g. HighCheeks) has different values of skew. Specifically, we created three versions of training data each with skew 0.5, 0.7, 0.9, while keeping the total number of images fixed. We trained a GAN on each training set, created a synthetic de-biased dataset with our method, and trained an attribute classifier with the training set and 160,000 pairs of synthetic images. For comparison, we also trained baseline models on just the differently skewed training sets. The classifiers were evaluated on the CelebA validation set. Table 10 summarizes the results. Compared to the baseline, our model has lower AP as expected, better DEO for skew 0.5 and 0.7, worse DNAP, and better or on par BA. Overall, classifiers trained on more imbalanced data with higher skew perform worse on all metrics.

C.2 Discriminability of attributes

Nam et al. [30] recently observed that correlations among attributes affect a classifier only if the protected attribute is 'easier' to learn than the target attribute. Inspired by their observation, we design an experiment where we put a pair of CelebA attributes in competition to assess their relative discriminability. We create a fully skewed dataset in which half of the images have both attributes and the other half

Attribute type	Attribute statistics			Hyperplane acc.		Hyperplane AP		Improvement over baseline			
Inconsistently labeled	Positive	Skew		g=−1	g=−1	g=−1	g=1	AP	DEO	BA	KL
`BigLips`	24.1%	0.73	g=−1	80.3	92.0	49.7	28.9	-0.35	-0.79	1.23	-0.03
`BigNose`	23.6%	0.75	g=1	91.7	74.5	51.1	82.4	-0.66	11.03	2.52	1.04
`OvalFace`	28.3%	0.68	g=−1	75.4	74.2	85.3	63.1	-1.82	7.53	3.33	0.77
`PaleSkin`	4.3%	0.76	g=−1	94.4	96.9	48.4	30.9	-1.90	4.26	0.31	0.26
`StraightHair`	20.9%	0.52	g=−1	87.7	69.8	25.0	58.8	-1.76	0.94	0.53	-0.08
`WavyHair`	31.9%	0.81	g=−1	73.0	92.1	79.4	23.5	-0.65	7.59	1.33	0.26
Gender-dependent	Positive	Skew		g=−1	g=1	g=−1	g=1	AP	DEO	BA	KL
`ArchedBrows`	26.6%	0.92	g=−1	72.3	92.1	82.6	25.5	-0.69	-3.31	-0.09	0.02
`Attractive`	51.4%	0.77	g=−1	88.4	81.0	97.9	81.9	-0.33	3.25	0.98	0.41
`BushyBrows`	14.4%	0.71	g=1	94.5	79.6	37.6	62.0	-1.20	8.49	1.14	0.25
`PointyNose`	27.6%	0.75	g=−1	73.6	82.9	84.4	59.9	-1.32	3.25	0.99	-0.40
`RecedingHair`	8.0%	0.62	g=1	94.5	88.3	41.8	57.7	-1.44	2.32	0.40	0.17
`Young`	77.9%	0.66	g=−1	96.2	84.1	99.7	95.3	-0.24	0.78	0.49	0.31
Gender-independent	Positive	Skew		g=−1	g=1	g=−1	g=1	AP	DEO	BA	KL
`Bangs`	15.2%	0.77	g=−1	90.3	94.9	81.5	58.9	-0.50	0.62	0.38	0.09
`BlackHair`	23.9%	0.52	g=1	89.3	83.2	78.9	79.2	-1.00	2.25	0.44	0.00
`BlondHair`	14.9%	0.94	g=−1	88.9	97.1	82.7	19.8	-0.77	1.04	0.23	-0.12
`BrownHair`	20.3%	0.69	g=−1	66.4	80.4	45.5	38.8	-0.51	-0.57	-0.01	0.01
`Chubby`	5.8%	0.88	g=1	99.1	89.9	7.6	33.8	-1.95	4.08	0.01	0.13
`EyeBags`	20.4%	0.71	g=1	90.7	74.4	64.1	74.4	-1.74	8.30	1.91	0.58
`Glasses`	6.5%	0.80	g=1	97.8	92.5	60.3	77.8	-0.24	-0.07	0.05	-0.27
`GrayHair`	4.2%	0.86	g=1	98.4	92.6	10.4	32.9	-2.60	7.02	0.32	0.54
`HighCheeks`	45.2%	0.72	g=−1	86.3	86.3	95.2	83.5	-0.33	-1.06	0.24	0.04
`MouthOpen`	48.2%	0.63	g=−1	88.6	87.0	96.4	93.1	-0.08	0.69	0.34	-0.03
`NarrowEyes`	11.6%	0.56	g=−1	93.8	92.1	29.6	26.4	-0.97	3.10	-0.53	0.12
`Smiling`	48.0%	0.65	g=−1	91.5	90.7	98.0	96.5	-0.09	1.01	0.67	0.03
`Earrings`	18.7%	0.97	g=−1	71.8	96.3	56.9	3.0	-0.63	8.18	0.64	1.40
`WearingHat`	4.9%	0.70	g=1	97.4	94.0	45.0	60.6	-0.95	2.67	0.14	-0.06
Average	24.1%	0.73		87.4	86.9	62.9	55.7	-0.95	3.18	0.69	0.21

Table 8: Attribute-level information. The columns are (from left to right) target attribute name, percentage of positive samples, skew, hyperplane accuracy, hyperplane AP, and our model's improvement over the baseline model on the four evaluation metrics.

Attribute	Change	Attribute	Change
`ArchedBrows`	**0.314**	`Glasses`	0.109
`Attractive`	**0.336**	`GrayHair`	0.056
`Bangs`	0.120	`HighCheeks`	**0.233**
`BlackHair`	0.153	`MouthOpen`	0.187
`BlondHair`	0.180	`NarrowEyes`	0.066
`BrownHair`	0.158	`PointyNose`	0.152
`BushyBrows`	0.136	`RecedingHair`	0.069
`Chubby`	0.067	`Smiling`	0.176
`Earrings`	0.176	`WearingHat`	0.065
`Eyebags`	**0.212**	`Young`	**0.268**

Table 9: We report the average classifier score change in an attribute when trying to create balanced datasets for other attributes. Classifier scores are between 0 and 1, and changes above 0.2 are bolded. We find that some attributes (e.g. `Attractive`, `Young`) change by a lot, whereas others (e.g. `GrayHair`, `WearingHat`) do not change much.

Skew	AP ↑		DEO ↓	
	Base	Ours	Base	Ours
0.5	**95.1** ± **0.3**	93.6 ± 0.4	7.0 ± 1.7	**6.6** ± **1.8**
0.7	**94.8** ± **0.3**	94.1 ± 0.3	19.6 ± 1.9	**19.4** ± **1.9**
0.9	**94.1** ± **1.7**	93.1 ± 0.4	**31.3** ± **2.0**	32.9 ± 1.9
Skew	BA ↓		KL ↓	
	Base	Ours	Base	Ours
0.5	-1.9 ± 0.5	**-3.0** ± **0.5**	0.4 ± 0.1	**0.3** ± **0.1**
0.7	**3.4** ± **0.5**	**3.4** ± **0.5**	**0.9** ± **0.1**	**0.9** ± **0.1**
0.9	7.1 ± 0.5	**7.0** ± **0.5**	**1.7** ± **0.1**	1.9 ± 0.1

Table 10: Comparison of `HighCheeks` attribute classifiers trained on differently skewed data.

have neither. With this dataset, we train a classifier to predict if an image has both attributes or neither. At test time, we evaluate the classifier on a perfectly balanced subset of the CelebA validation set (where each of the four possible hat-glasses combinations occupies a quarter of the dataset), and compute AP for each attribute. If one attribute has a higher AP than the other, it suggests that this attribute is 'easier' to learn than the other. We repeat this experiment with a second dataset skewed in a different way (i.e. half of the images have one attribute but not the other).

The results for gender-dependent and gender-independent attributes are in Table 11. We report that an attribute is 'eas-

	ArchedBrows	Attractive	Bangs	BlackHair	BlondHair	BrownHair	BushyBrows	Chubby	Earrings	EyeBags	Glasses	GrayHair	HighCheeks	MouthOpen	NarrowEyes	PointyNose	RecedingHair	Smiling	WearingHat	Young
Gender	y	y	y	y	y	y	y	y	y	y	n	y	y	y	y	y	y	y	n	y
Glasses	y	y	y	y	y	y	y	y	y	y	–	y	y	y	y	y	y	y	y	y
Young	n	n	n	n	n	y	n	n	y	n	n	n	n	n	n	y	n	n	y	–

Table 11: Discriminability of attributes. We compare attributes on the row to those in the columns. y indicates that the attribute in the row is easier to learn than that in the column and n indicates the opposite. We find that gender expression is one of the easiest attributes to learn, while Young is relatively hard.

ier’ to learn than the other if it has a higher AP for both created datasets. We find that gender expression is one of the easiest attributes to learn, which may be why gender bias is prevalent in many models. On the other hand, Young is relatively hard for a model to learn, so its correlation with other attributes may not be as influential. We find that gender expression is one of the easiest attributes to learn (with gender expression having a higher AP than every attribute we tested except WearingHat and Glasses), which may be why gender bias is prevalent in many models. On the other hand, Young is relatively hard for a model to learn (Young is harder to learn than all but 4 other attributes), so its correlation with other attributes may not be as influential.

D. Ablation studies

In this section, we describe in more detail the ablation studies we have conducted to investigate how improved hyperplanes and use of different labels for synthetic images impact (or not) our method’s performance (Section 4.2 of the main paper).

We first investigate if hyperplanes estimated with better balanced samples improve the performance of downstream attribute classifiers. We test this hypothesis by training models using hyperplanes that are estimated with different fractions of positive or negative samples.

For the attribute HighCheeks, we estimate hyperplanes with different fractions of positive and negative samples, while keeping the total number of samples constant at 12,000 and the number of positive samples same for each gender expression. We then train attribute classifiers with the CelebA training set and synthetic pair images augmented with these different hyperplanes. In Table 12, we report results evaluated on the CelebA validation set. We find that although the fairness metrics deteriorate as the target attribute hyperplanes were estimated with less balanced samples, this rate is relatively slow, and the downstream classifier still performs reasonably well.

Next, we tried training models with synthetic images with the same hallucinated target labels, i.e. using only $G(\mathbf{z})$ and $G(\mathbf{z}')$ such that $f_t(G(\mathbf{z}))=f_t(G(\mathbf{z}'))$, and labeling synthetic images with $h_t(\mathbf{z})$ in place of $f_t(G(\mathbf{z}))$. Table 13 contains all results. We report average results over all gender-dependent and gender-independent attributes. We find that both these ablations are comparable to ours, with in a slight loss in AP (79.8 and 82.1 versus 82.6), and worse fairness metrics in general (average DEO is 18.1 and 17.4 vs 16.1, BA is 0.9 and 0.7 vs 0.5).

Fraction	AP ↑	DEO ↓	BA ↓	KL ↓
50.0%	95.1 ± 0.3	13.2 ± 1.7	0.5 ± 0.5	0.7 ± 0.1
12.5%	95.1 ± 0.3	14.0 ± 1.7	0.8 ± 0.5	**0.6 ± 0.1**
6.3%	95.1 ± 0.3	15.1 ± 1.8	1.3 ± 0.5	0.8 ± 0.2
3.1%	95.1 ± 0.3	14.2 ± 1.7	1.0 ± 0.5	0.7 ± 0.1
1.6%	95.1 ± 0.3	**12.9 ± 1.8**	**0.3 ± 0.5**	0.7 ± 0.1

Table 12: The amount of underrepresentation in samples used for hyperplane estimation doesn’t appear to affect the performancee of the downstream classsification model much.

	AP ↑	DEO ↓	BA ↓	KL ↓
$f_t(G(\mathbf{z})) = f_t(G(\mathbf{z}'))$	79.8 ± 1.6	17.4 ± 4.5	0.9 ± 0.4	**1.0 ± 0.3**
Labels computed using h_t	82.1 ± 1.5	18.1 ± 4.2	0.7 ± 0.4	1.4 ± 0.8
Ours	**82.6 ± 1.5**	**16.1 ± 4.2**	**0.5 ± 0.4**	1.3 ± 0.7

Table 13: Mean performances over all gender-dependent and gender-independent attributes on the validation set when using different methods to pick and label synthetic images. We find that most performances are comparable, with our method having a slightly higher AP, and slightly better DEO and KL.

Metric	Num. of samples used to compute f_g				
	10	100	1000	10000	162,770
AP ↑	78.8 ± 1.5	78.8 ± 1.5	78.8 ± 1.5	**78.9 ± 1.6**	78.7 ± 1.6
DEO ↓	11.1 ± 3.4	11.3 ± 3.0	10.5 ± 3.7	10.8 ± 3.7	**9.6 ± 3.1**
BA ↓	0.6 ± 0.5	1.0 ± 0.5	0.5 ± 0.5	0.7 ± 0.5	**0.4 ± 0.5**
KL ↓	0.6 ± 0.2	0.8 ± 0.3	0.7 ± 0.3	0.7 ± 0.3	**0.5 ± 0.6**

Table 14: Average over 4 attributes when using different numbers of labeled examples to compute gender expression. Results are reported on the validation set. We find that while the fairness metrics improve slightly by using more labelled examples, this is gradual, and within the error bars, in all cases.

E. Number of required labeled images

Choi et al. [12] use a method that is unsupervised. Assuming access to a small unbiased dataset, as well as a large (possibly biased) dataset, they estimate the bias in the larger dataset, and learn a generative model that generates unbiased data at test time. Using these generated images, as well as real images, they train a downstream classifier for the attribute Attractive, and achieve an accuracy of 75%. Since most of the protected attributes that we care about are sensitive (for example gender or race), not requiring protected attribute labels prevents perpetuation of harmful stereotypes. In order to understand how much our model depends on the protected attribute labels, we investigate where our model depends on the protected attributes labels. We use protected attribute labels only to compute the linear separator in the latent space ($\mathbf{w_g}$ and b_g from section A in this document). We now train classifiers for gender expression, using different numbers of labeled images, and use these classifiers to train target attribute classifiers for 4 different attributes (EyeBags, BrownHair, GrayHair and HighCheeks). Most of the fairness metrics improve slightly when using more labeled examples (DEO improves from 11.1 when using just 10 samples to 9.6 when using all 162k samples in the CelebA training set, BA improves from 0.6 to 0.4, and KL improves from 0.6 to 0.5), however, these are all gradual, and within the error bars. Full results are in Table 14.