Discover the Unknown Biased Attribute of an Image Classifier

Zhiheng Li Chenliang Xu
University of Rochester
{zhiheng.li,chenliang.xu}@rochester.edu

Abstract

Recent works find that AI algorithms learn biases from data. Therefore, it is urgent and vital to identify biases in AI algorithms. However, the previous bias identification pipeline overly relies on human experts to conjecture potential biases (e.g., gender), which may neglect other underlying biases not realized by humans. To help human experts better find the AI algorithms' biases, we study a new problem in this work – for a classifier that predicts a target attribute of the input image, discover its unknown biased attribute.

To solve this challenging problem, we use a hyperplane in the generative model's latent space to represent an image attribute; thus, the original problem is transformed to optimizing the hyperplane's normal vector and offset. We propose a novel total-variation loss within this framework as the objective function and a new orthogonalization penalty as a constraint. The latter prevents trivial solutions in which the discovered biased attribute is identical with the target or one of the known-biased attributes. Extensive experiments on both disentanglement datasets and real-world datasets show that our method can discover biased attributes and achieve better disentanglement w.r.t. target attributes. Furthermore, the qualitative results show that our method can discover unnoticeable biased attributes for various object and scene classifiers, proving our method's generalizability for detecting biased attributes in diverse domains of images.

1. Introduction

Although the performance of deep neural networks is greatly improved by training on large-scale datasets, worrisome biases are also learned by AI algorithms. Thus it is imperative to identify AI algorithms' biases, whereas the previous bias identification pipeline [5, 39] has some shortcomings. First, it overly relies on human experts to speculate potential biases (Step 1 in Fig. 1 (a)), which may leave other unconsidered biases unexposed. For example, people may conjecture legally protected attributes (*e.g.*, *age*, *gender*[1]) for face image classifiers and consider "scene bias" for the action recognition task [7]. However, people may neglect other unnoticeable biased attributes such as *hair length* [3] and the *presence of children* [53] spuriously correlated with *gender*. Although they are not legally protected attributes, not considering these biased attributes may still lead to unfairness against different genders [53]. Second, the previous pipeline also needs expensive human efforts to collect testing images and annotate biased attributes (Step 2 in Fig. 1 (a)) for analyzing classifier's predictions. When one wants to analyze biases in a new domain of images (*e.g.*, object

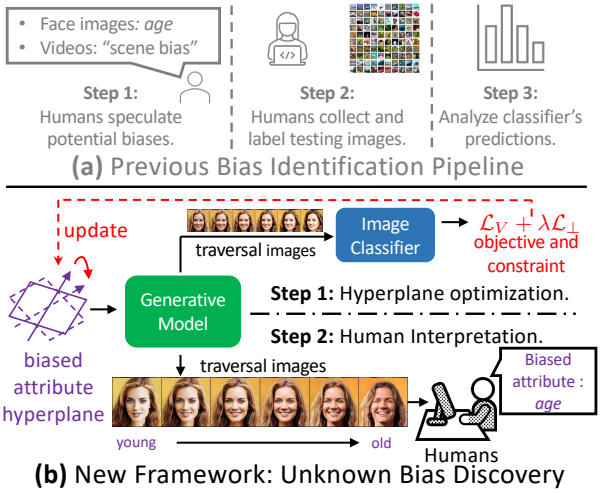

Figure 1: While the previous pipeline (a) overly relies on human efforts to identify biases, we devise a new automated framework **(b)** that helps humans discover the unknown biased attribute of an image classifier. In step 1, a generative models' biased attribute hyperplane in the latent space is optimized by our proposed optimization objective and constraints. In step 2, humans can interpret the semantic meaning of the biased attribute hyperplane from the transformation in the synthesized traversal images. For example, images changing from "young" to "old" imply the biased attribute *age*.

The code is available at https://git.io/J3kMh.

[1] In this paper, we use *gender* to denote visually perceived gender, which does not indicate the person's true gender identity.

and scene categories in ImageNet [11] and Place365 [56]), massive human efforts of image collection and labeling are needed for each new image domain, which is not scalable. In addition, as a down-stream task of bias identification, many de-biasing methods [7, 10, 45, 54] also require well-defined biased attributes and annotations as inputs and supervisions to mitigate corresponding biases. As a result, if the previous pipeline does not identify the biases due to either negligence of biases or limited annotation budget, the biases will not be mitigated by those de-biasing methods. Therefore, it is urgent to discover unknown AI biases with less human effort.

To this end, we study a novel problem by defining the *unknown biased attribute discovery task*: for a classifier that predicts a target attribute of the input image, discover its *unknown* biased attribute. The "target attribute" stands for the attribute for prediction. The "biased attribute" means an attribute that violates the fairness criteria [18, 31] and differs from the target attribute. For example, if a gender classifier has different predictions over female images of different skin colors, then the *skin tone* attribute is the biased attribute. The "*unknown*" has two levels of meanings. First, it indicates that the biased attribute that is expected to be discovered is *not* presumed by humans, not to mention annotating images with the biased attribute labels. That is, the traditional pipeline in Fig. 1 (a) does *not* meet the requirement of "*unknown*." Second, "*unknown*" also implies that human experts may have already known some biased attributes and expect a different one. After completing this task, the discovered biased attributes can be used as inputs for other down-stream tasks, such as algorithmic de-biasing.

We propose a novel framework (Fig. 1 (b)) for this new task by solving two challenges. The first difficulty is how to represent and learn the "attribute" without any presumptions or labels. To tackle this problem, we base our method on some findings in [23, 46, 48] that the hyperplane in the latent space of generative models can linearly separate an attribute's values. Hence, we represent the unknown biased attribute as an optimizable hyperplane in a generative model's latent space (see "biased attribute hyperplane" in Fig. 1 (b)). Different from previous methods [23, 46, 48] using labels of attribute as the supervision, we propose *total variation loss* ($\mathcal{L}_V$ in Fig. 1 (b)) to optimize the hyperplane that induces the violation of the fairness criterion, without requiring any attribute labels. The second challenge is how to ensure the discovered biased attribute is different from the known ones. We propose *orthogonalization penalty* ($\mathcal{L}_\perp$ in Fig. 1 (b)) to encourage disentanglement between the biased attribute and known attributes. We also use $\mathcal{L}_\perp$ to prevent the biased attribute from being identical with the classifier's target attribute. Finally, to enable humans to interpret the semantic meaning of the optimized hyperplane, a sequence of images, dubbed as traversal images, are generated based on the optimized hyperplane. The variation along the traversal images is the semantic meaning of the optimization result. As shown in Fig. 1, the synthesized traversal face images gradually transform from "young" to "old," indicating that the biased attribute found by our method is *age*. In summary, compared with the previous pipeline (Fig. 1 (a)), our framework first lets the optimization actively find the biased attribute (Step 1 in Fig. 1 (b)) and postpones human involvement to the final step (Step 2 in Fig. 1 (b)), which not only automatically discovers the unknown biases that human may not realize, but also exempts human efforts from annotating biased attributes on testing images.

We conduct three experiments to verify the effectiveness of our method. In the first experiment, two disentanglement datasets [20, 32] are used for creating large-scale experimental settings for evaluation. In the second experiment, we conduct experiments on two face datasets [23, 34] for discovering biased attributes in face attribute classifiers. The first two experiments show that our method can correctly discover the biased attribute. In the third experiment, we apply our method for discovering the biased attribute in other domains of images, such as objects and scenes. The qualitative results and the user study show that our method can discover unnoticeable biases from classifiers pretrained on ImageNet [11] and Place365 [56], proving our method's generalizability for finding biases in various image domains.

The contributions of this work are as follows. First, we propose a novel *unknown biased attribute discovery task* for discovering unknown biases from classifiers. Solving the problems in this task can help humans better identify classifiers' biases. Second, we propose a novel method for this new task by optimizing the *total variation loss* and the *orthogonalization penalty* without any presumptions or labels of biased attributes. Lastly, we design comprehensive experiment settings and evaluation metrics to verify the effectiveness of our method, which can also be used as benchmarks for future works. Furthermore, many related fields can be benefited from our new framework for discovering unknown biases, such as algorithmic de-biasing, dataset audition, *etc*. (more discussions in Appendix H.5).

2. Related Work

Bias Identification The previous bias identification pipeline mainly focuses on collecting testing images and analyzing the performances in different subgroups based on biased attribute value. Buolamwini and Gebru [5] collect in-the-wild face images to analyze the error rates discrepancies in intersectional subgroups. Kortylewski *et al.* [28, 29] use 3DMM [4] to synthesize 3D face images in different poses and lighting conditions. Muthukumar *et al.* [40] alter facial attributes of face images via image processing techniques such as color theoretic methods and image cropping. Denton *et al.* [12, 13] use PGGAN [22] to synthesize images with different values of attributes in CelebA dataset [34]. To

further reduce the correlation between attributes, Balakrishnan *et al.* [3] additionally annotate attributes of synthesized image to generate multi-dimensional "transects" based on StyleGAN2 [24], where each dimension only changes the value of one attribute and remains other attributes unchanged. To reduce the dependency on the human labels, Donderici *et al.* [14] use a physics engine to synthesize face images by controlling different facial attributes values then augment the images to the real-world domain. Without relying on a specific algorithm for bias analysis, Wang *et al.* [52] propose "REVISE," a tool for computing datasets' statistics in terms of object, gender, and geography. Manjunatha *et al.* [38] analyze biases of VQA [2] task by running rule mining algorithms. All of the methods above can only analyze the algorithm's biases from a presumed set of attributes or annotations. In contrast, we study a novel problem on discovering the *unknown* biased attribute.

Bias Mitigation Many methods have been proposed to mitigate AI algorithms' biases, and most of them require the supervision of protected attributes. Wang *et al.* [54] benchmark previous bias mitigation methods with full-supervision of protected attributes. Creager *et al.* [10] train a VAE-based disentanglement method from protected attributes' labels so that the learned representation can be flexibly fair to multiple protected attributes during the testing time. Sarhan *et al.* [45] propose a method to maximize the entropy of the protected attribute's prediction and orthogonalize the mean vectors of normal distributions of target attribute and protected attributes. Vowels *et al.* [50] propose a weakly-supervised bias mitigation method, NestedVAE, which is trained with paired images from different protected attribute values. Choi *et al.* [8] use the weak supervision from a small reference dataset with balanced distribution to mitigate the biases. Though different levels of supervision are used, all of the works mentioned above require the selected protected attributes as inputs. The only exception is [35], where neither definition of the protected attribute nor the labels are required, and they prove that better disentanglement can decrease the unfairness score. However, the experiments in [35] are only based on synthetic datasets with balanced distribution. In comparison, our method is tested to be effective on real-world datasets.

Unsupervised Disentanglement The disentanglement methods aim to recover different independent attributes (*i.e.*, factors of variations) from data by learning a generative model. We relate this field to our work because unsupervised disentanglement methods, which factorize attributes of data without any definitions or labels, can be used as baseline methods for the *unknown biased attribute discovery task*. For VAE [27]-based generated models, many methods, including β-VAE [20], FactorVAE [25], β-TCVAE [6], DIP-VAE-I and DIP-VAE-II [30] JointVAE [15], are proposed for unsupervised disentanglement, where the goal is to represent each attribute as one dimension of the hidden space of VAE. For GAN-based generative models, Voynov and Babenko [51] train an additional reconstructor to predict the direction index and shift magnitude. More recently, Hessian Penalty [44] disentangles factors of variations by penalizing the off-diagonal items in the hessian matrix w.r.t. latent code. We use these unsupervised disentanglement methods as baselines to investigate their performances on the *unknown biased attribute discovery task*.

3. Unknown Biased Attribute Discovery Task

In this section, we initially introduce the definition of fairness in Sec. 3.1. Then we formally define the *unknown biased attribute discovery task* in Sec. 3.2.

3.1. Fairness Definition

In this work, we focus on the counterfactual fairness criterion [12, 13, 21, 31] in the image domain and we leave studies on other fairness criteria in future works. The counterfactual fairness is formulated by:

$$P(\hat{\mathsf{t}} \mid \mathbf{I}(\mathsf{s}=s_1)) = P(\hat{\mathsf{t}} \mid \mathbf{I}(\mathsf{s}=s_2)), s_1 \neq s_2, \quad (1)$$

where t is the target attribute. s_1 and s_2 are different values of the protected attribute (also called sensitive attribute) s. $\mathbf{I}(\mathsf{s}=s_1)$ and $\mathbf{I}(\mathsf{s}=s_2)$ are pair of counterfactual images intervened in terms of the protected attribute s by assigning different protected attribute values: s_1 and s_2. The values of all other attributes, including the target attribute, are the same between two images. $P(\hat{\mathsf{t}} \mid \mathbf{I})$ is a classifier's prediction of the target attribute t of the image $\mathbf{I}$. For example, if s_1 and s_2 are "young" and "old" when the protected attribute is *age* and the target attribute is *gender*, then $\mathbf{I}(\mathsf{s}=s_1)$ and $\mathbf{I}(\mathsf{s}=s_2)$ are two images of the same person in different ages. The counterfactual fairness criterion requires the classifier's *gender* prediction to be identical between two images.

3.2. Formulation of Bias Attribute Discovery Task

Here, we formally define the *unknown biased attribute discovery task*. The input of this task is a classifier for predicting a target attribute t of the input image $\mathbf{I}$ (*i.e.*, $P(\hat{\mathsf{t}} \mid \mathbf{I})$). At the same time, the classifier also learns unknown biases from its training data. We formulate such bias in the classifier as the unknown biased attribute b that **violates** the fairness criterion:

$$P(\hat{\mathsf{t}} \mid \mathbf{I}(\mathsf{b}=b_1)) \neq P(\hat{\mathsf{t}} \mid \mathbf{I}(\mathsf{b}=b_2)), b_1 \neq b_2, \quad (2)$$

where b_1 and b_2 are different values of the biased attribute. Eq. 2 means that the predictions of the target attribute are correlated the biased attribute. The expected output of this task is the biased attribute b. In other words, the unknown biased attribute b should be discovered. Additionally, a set of known attributes $K = \{\mathsf{k}\}$ can be provided for requiring

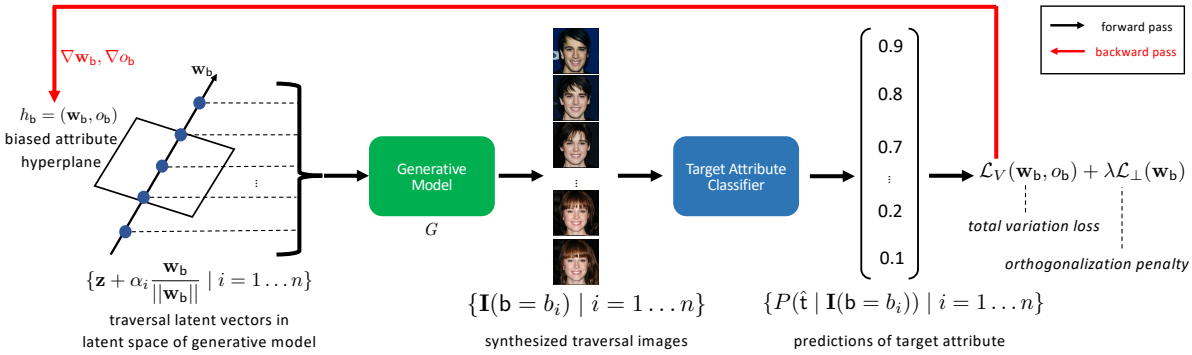

Figure 2: Method overview. We use a latent space hyperplane to represent the biased attribute. During training, we first sample the traversal latent vectors along the normal vector $\mathbf{w}_\mathsf{b}$ of the biased attribute hyperplane h_b in the latent space of the generative model G to synthesize the traversal images, which are then are fed into the target attribute classifier. Finally, we use classifier's predictions for all traversal images to compute *total variation loss* ($\mathcal{L}_V$), which is jointly minimized with *orthogonalization penalty* ($\mathcal{L}_\perp$) for optimizing the hyperplane h_b. The weights of the generative model and the classifier are fixed.

that the discovered biased attribute should not be one of known attributes. The set of known attributes is useful when a user have already known some biases that the classifier has and expect to know other unknown biased attributes of the classifier. Providing the known attribute is optional (*i.e.*, $K = \varnothing$) when the user has not known any biased attributes.

4. Method

In this section, we present our method on *unknown biased attribute discovery task*. The overview of our method is shown in Fig. 2. First, we represent attributes of images as the hyperplanes in the latent space of the generative model (see the left side of Fig. 2). Then, we formalize our method as an optimization problem and propose *total variation loss* to optimize the hyperplane of the biased attribute (see Sec. 4.2). To avoid some unwanted results, we propose *orthogonalization penalty* in Sec. 4.3 as the constraints for the optimization problem. Finally, we summarize the full model in Sec. 4.4.

4.1. Representation of the Biased Attribute

As defined in Sec. 3.2, an unknown biased attribute b needs to be discovered in *unknown biased attribute discovery task*. We approach this task by solving an optimization problem. Therefore, we need to formulate the biased attribute as an optimizable representation. To this end, we leverage a generative model G (Generative Model in Fig. 2) that synthesizes the image $\mathbf{I}$ from a latent vector $\mathbf{z} \in \mathbb{R}^d$, where d denotes the dimensions of G's latent space. Here G can be implemented by the generator of GAN [16] or the decoder of VAE [27]. Some recent works in the field of image editing [46, 48] find that the hyperplane in generative model's latent space can be learned with full-supervision of attribute labels to linearly separate attribute values. Based on this finding, we represent the biased attribute as the hyperplane $h_\mathsf{b} = (\mathbf{w}_\mathsf{b}, o_\mathsf{b})$ in G's latent space (left side of Fig. 2), where $\mathbf{w}_\mathsf{b} \in \mathbb{R}^d$ and $o_\mathsf{b} \in \mathbb{R}$ are normal vector and offset of a's hyperplane. In this way, the problem of discovering the biased attribute can be transformed into an optimization problem by learning the hyperplane h_b.

4.2. Total Variation Loss

After formalizing the biased attribute as an optimizable representation, the next question is how to design an optimization objective. Note that, different from image editing task [46, 48] where attribute labels are available as full supervision, we do not have labels for learning the hyperplane because the biased attribute is even unknown, not to mention collecting labels for supervised training.

To solve this challenging problem, we utilize the definition in Sec. 3.2 that the unknown biased attribute violates the fairness criterion. In order to check if the hyperplane h_b violates the fairness criterion, we generate N images that have different values of the biased attribute, formulated as $\{\mathbf{I}(\mathsf{b} = b_i) \mid i = 1 \dots N\}$ (shown in the middle of Fig. 2). We term these images as traversal images. We achieve this by the following steps. First, we randomly sample a latent vector $\mathbf{z}$. Then, since the normal vector $\mathbf{w}_\mathsf{b}$ of the hyperplane h_b is the most discriminative direction to separate different values of b, we traverse along the normal vector starting from latent vector $\mathbf{z}$, resulting in traversal latent vectors $\{\mathbf{z} + \alpha_i \frac{\mathbf{w}_\mathsf{b}}{||\mathbf{w}_\mathsf{b}||} \mid i = 1 \dots N\}$ (illustrated as blue dots in Fig. 2), where α_i is the i-the step size of the traversal. Finally, the traversal images can be synthesized by feeding the traversal latent vectors into the generative model G (*i.e.*, $\mathbf{I}(\mathsf{b} = b_i) = G(\mathbf{z} + \alpha_i \frac{\mathbf{w}_b}{||\mathbf{w}_b||})$).

After synthesizing the traversal images, we feed them to the classifier ("Target Attribute Classifier" in Fig. 2) and obtain the target attribute predictions of the traversal images $\{P(\hat{\mathsf{t}} \mid \mathbf{I}(\mathsf{b} = b_i)) \mid 1 \dots N\}$. Then we propose the *total variation loss* ($\mathcal{L}_V$) as the objective function, which quantifies the degree of violation against the fairness definitions:

$$\mathcal{L}_V = -\log \frac{1}{N-1} \sum_{i=1}^{N-1} |P(\hat{\mathtt{t}} \mid G(\mathbf{z} + \alpha_{i+1} \frac{\mathbf{w}_\mathtt{b}}{||\mathbf{w}_\mathtt{b}||})) - P(\hat{\mathtt{t}} \mid G(\mathbf{z} + \alpha_i \frac{\mathbf{w}_\mathtt{b}}{||\mathbf{w}_\mathtt{b}||}))|. \quad (3)$$

Intuitively, *total variation loss* checks the fairness definition of each consecutive predictions in $\{P(\hat{\mathtt{t}} \mid \mathbf{I}(\mathtt{b} = b_i)) \mid 1 \dots N\}$, and larger differences over the predictions lead to lower *total variation loss*. Since all aforementioned operations are differentiable, we minimize the *total variation loss* by updating the hyperplane via gradient decent. In practice, offset $o_\mathtt{b}$ is used for projecting the sampled latent vector $\mathbf{z}$ to the hyperplane, so we optimize both $\mathbf{w}_\mathtt{b}$ and $o_\mathtt{o}$ by $\mathcal{L}_V$. The complete algorithm for computing $\mathcal{L}_V$ is in Appendix A.2.

4.3. Orthogonalization Penalty

However, minimizing *total variation loss* ($\mathcal{L}_V$) alone has two problems. First, it may lead to a trivial solution where the discovered biased attribute is just the target attribute because images with different target attribute values will definitely produce large variations in the target attribute predictions. For example, a *gender* classifier will have large prediction variations when the traversal images transform from "male" to "female." Secondly, the task allows users to provide a set of known attribute $K = \{\mathtt{k}\}$ and the discovered biased attribute should not be one of the known attribute $\mathtt{k}$ (see Sec. 3.2), which cannot be achieved by minimizing $\mathcal{L}_V$. To tackle these two problems, we also represent the target attribute $\mathtt{t}$ and known attribute $\mathtt{k}$ as hyperplanes in the latent space, denoted by $h_\mathtt{t} = (\mathbf{w}_\mathtt{t}, o_\mathtt{t})$ and $h_\mathtt{k} = (\mathbf{w}_\mathtt{k}, o_\mathtt{k})$, respectively. These normal vectors and offsets can be obtained through supervised training because the target attribute and the known attributes are pre-defined. More details of how to get these hyperplanes are shown in Appendix A.3. Then, we propose the *orthogonalization penalty* ($\mathcal{L}_\perp$) to tackle two problems mentioned above:

$$\mathcal{L}_\perp = \mathbf{w}_\mathtt{b}^T \mathbf{w}_\mathtt{t} + \sum_{\mathtt{k} \in K} \mathbf{w}_\mathtt{b}^T \mathbf{w}_\mathtt{k}. \quad (4)$$

Minimizing $\mathcal{L}_\perp$ encourages the biased attribute's hyperplane $h_\mathtt{b}$ to be orthogonalized with hyperplanes of the target attribute and known attributes. Intuitively, better orthogonalization will produce a smaller variation of traversal latent vectors' projections onto $h_\mathtt{t}$ and $h_\mathtt{k}$.

4.4. Full Model

Finally, we jointly minimize the *total variation loss* and the *orthogonality penalty* to update the hyperplane $h_\mathtt{b} = (\mathbf{w}_\mathtt{b}, o_\mathtt{b})$ (see red line in Fig. 2):

$$\mathcal{L} = \mathcal{L}_V + \lambda \mathcal{L}_\perp \,, \quad (5)$$

where λ is a coefficient of the *orthogonality penalty*.

5. Experiment

The experiments are conducted on disentanglement datasets (Sec. 5.1), face images (Sec.5.2), and images from other domains (*e.g.*, cat, bedroom, *etc.*) (Sec. 5.3). More details of experimental settings on each dataset will be introduced in each subsection. Additional implementation details can be seen in Appendix (Appx.) A.

Evaluation Metrics As introduced in Sec. 4, we use hyperplane in the latent space to represent an attribute. For quantitative evaluation, we first choose a pair of different attributes as the ground-truth biased attribute and target attribute. Then we compute the ground-truth hyperplanes of these two attributes (more details in Appx. A.3). Based on the normal vectors of hyperplanes, we design the following *quantitative* evaluation metrics:

1. $|\cos\langle \hat{\mathbf{w}}_\mathtt{b}, \mathbf{w}_\mathtt{b} \rangle|$ is the absolute value of cosine similarity between the predicted normal vector $\hat{\mathbf{w}}_\mathtt{b}$ and the ground-truth normal vector $\mathbf{w}_\mathtt{b}$ of biased attribute's hyperplanes. Larger $|\cos\langle \hat{\mathbf{w}}_\mathtt{b}, \mathbf{w}_\mathtt{b} \rangle|$ implies that the hyperplane prediction is closer to the ground-truth biased attribute.
2. $|\cos\langle \hat{\mathbf{w}}_\mathtt{b}, \mathbf{w}_\mathtt{t} \rangle|$ is the absolute value of cosine similarity between the predicted normal vector $\hat{\mathbf{w}}_\mathtt{b}$ of the biased attribute's hyperplane and the ground-truth normal vector $\mathbf{w}_\mathtt{t}$ of the target attribute's hyperplane. Lower value of $\cos\langle \hat{\mathbf{w}}_\mathtt{b}, \mathbf{w}_\mathtt{t} \rangle$ means that the hyperplane prediction is more orthogonal to the target attribute hyperplane. We refer to it as "better disentanglement w.r.t. the target attribute."
3. $\Delta\cos = |\cos\langle \hat{\mathbf{w}}_\mathtt{b}, \mathbf{w}_\mathtt{b} \rangle| - |\cos\langle \hat{\mathbf{w}}_\mathtt{b}, \mathbf{w}_\mathtt{t} \rangle|$ is difference of first two metrics. Larger values imply better results by jointly considering the first two metrics. **We use $\Delta\cos$ as the major evaluation metric** for comparing different methods because a good biased hyperplane prediction should simultaneously be closed to the ground-truth biased attribute hyperplane (*i.e.*, large $|\cos\langle \hat{\mathbf{w}}_\mathtt{b}, \mathbf{w}_\mathtt{b} \rangle|$) and be orthogonal to the target attribute hyperplane (*i.e.*, small $|\cos\langle \hat{\mathbf{w}}_\mathtt{b}, \mathbf{w}_\mathtt{t} \rangle|$).
4. "%leading": since each experiment contains multiple results under different settings (see **Experiment Settings** in Sec. 5.1), we report "%leading" of a method to denote the percentage of the number of settings that this method leads in terms of $\Delta\cos$.

For the *qualitative* evaluation metric, we show traversal images of different biased attribute hyperplane predictions based on the *same* sampled latent code. The traversal images of the accurate biased attribute hyperplane prediction will only have variations in terms of the ground-truth biased attribute. In other words, there exists no or relatively small variations in terms of the target attribute, known attribute, or any other attributes (see examples in Fig. 3).

5.1. Experiment on Disentanglement Datasets

Datasets In this experiment, we conduct experiments on two disentanglement datasets: SmallNORB [32] and

	method	$\lvert\cos\langle\hat{\mathbf{w}}_\mathbf{b}, \mathbf{w}_\mathbf{b}\rangle\rvert \uparrow$	$\lvert\cos\langle\hat{\mathbf{w}}_\mathbf{b}, \mathbf{w}_\mathbf{t}\rangle\rvert \downarrow$	$\Delta\cos \uparrow$	%leading ↑
SmallNORB	VAE-based	0.21±0.21	0.16±0.13	0.05±0.20	16.67%
	$\mathcal{L}_H$	**0.24±0.16**	0.26±0.16	-0.02±0.24	**31.67%**
	Ours	0.23±0.18	**0.10±0.11**	**0.12±0.21**	**51.67%**
dSprites	VAE-based	0.11±0.14	0.13±0.14	-0.01±0.16	22.00%
	$\mathcal{L}_H$	**0.23±0.15**	0.25±0.15	-0.02±0.21	**41.00%**
	Ours	0.17±0.14	**0.13±0.11**	**0.05±0.18**	**37.00%**

Table 1: The mean and standard deviation results averaged over all 480 experiment settings on SmallNORB [32] and dSprites [20] datasets. $\mathcal{L}_H$ denotes Hessian Penalty method. Top-2 results under %leading metric are bolded. ↑: larger value means better result. ↓: smaller value means better result. Note that $\Delta\cos$ is the major evaluation metric that jointly considers the first two metrics. Our method achieves better performance than two baseline methods.

dSprites [20]. Both datasets contain images with a finite set of attributes, such as *scale*, *shape*, *etc*. The numbers of attributes for two datasets are 4 and 5, respectively. We preprocess the attribute if it is not binary-valued or continuous-valued (*e.g.*, *shape*, *category*), and more details of preprocessing is shown in Appx. A.7.

Generative Models We choose 5 VAE-based methods as the generative models: vanilla VAE [27], β-VAE [20], β-TCVAE [6], DIP-VAE-I, and DIP-VAE-II [30]. We use the same set of hyperparameters reported in [36] (more details in Appx. A.1). The weights of the trained generative model is fixed and will not be updated when optimizing $h_\mathbf{b}$.

Baseline Methods

VAE-based: since the aforementioned VAE-based generative models are also disentanglement methods, we use them as baseline methods. Note that these methods directly disentangle dimensions of the latent space, meaning that the normal vector of the predicted hyperplane of the biased attribute is aligned with the axis and the offset of the hyperplane is 0.

Hessian Penalty [44] ($\mathcal{L}_H$) is another baseline method. We use the officially released implementation of the Hessian Penalty for optimizing the hyperplanes in the latent space.

More details of how to adapt baseline methods for *unknown biased attribute discovery task* are described in Appx. A.6.

Experiment Settings We create 480 experiment settings on two disentanglement datasets. Each experiment setting is a triplet of (target attribute, biased attribute, generative model) (*e.g.*, (*shape*, *scale*, β-VAE)). In each setting, to make sure the target attribute classifier is biased by the chosen biased attribute, we train it on a sampled dataset with a skewed distribution between the target attribute and the biased attribute. In the example of (*shape*, *scale*, β-VAE) setting on dSprites dataset, the skewed training set contains more "large heart" images than "small square" images. More details are described in Appendix A.8. For *orthogonalization penalty*, we choose all remaining attributes other than biased or target attributes as the "known attributes."

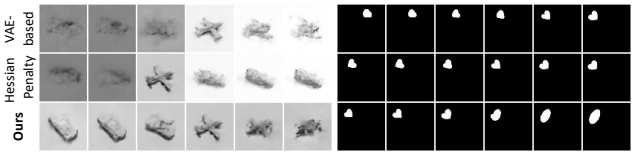

Figure 3: Qualitative comparison of traversal images on (a) SmallNORB [32] and (b) dSprites [20]. The rows are traversal images based on predicted hyperplanes from different methods. The ground-truth biased attributes of (a) and (b) are *category* and *shape*, respectively. The target attributes of (a) and (b) are *azimuth* and *position x*, respectively. The traversal images from the baseline methods vary in terms of the *lighting* in (a), *orientation* and *position x* in (b), which are different from the ground-truth biased attribute. In contrast, the traversal images from our method correctly vary in terms of the ground-truth biased attribute *category* (*i.e.* from "car" to "plane") in (a) and *shape* (from "heart" to "ellipse") in (b). The quantitative results (*e.g.*, $\Delta\cos$) of the experiment settings in this figure are in Appx. E.

Results For quantitative comparison, we report the mean and standard deviation of results over all experiment settings on each dataset. All results are summarized in Tab. 1. Surprisingly, the Hessian Penalty method achieves the best performance in terms of $\lvert\cos\langle\hat{\mathbf{w}}_\mathbf{b}, \mathbf{w}_\mathbf{b}\rangle\rvert$. However, it also achieves the worse performance in $\lvert\cos\langle\hat{\mathbf{w}}_\mathbf{b}, \mathbf{w}_\mathbf{t}\rangle\rvert$, implying that Hessian Penalty method learns a hyperplane that is averaged between the biased attribute and target attribute. Our method can achieve comparable results with the Hessian Penalty method in $\lvert\cos\langle\hat{\mathbf{w}}_\mathbf{b}, \mathbf{w}_\mathbf{b}\rangle\rvert$, and achieve the best result in $\lvert\cos\langle\hat{\mathbf{w}}_\mathbf{b}, \mathbf{w}_\mathbf{t}\rangle\rvert$. In terms of the major metric $\Delta\cos$ that jointly considers the first two metrics, three out of four results of the baseline methods are even negative values, meaning that their predicted hyperplanes are even closer to the target attribute hyperplanes. In contrast, our method achieves the best $\Delta\cos$ results on two datasets. In terms of %leading, our method can also achieve the best result on SmallNORB and comparable result with Hessian Penalty on dSprites. Note that our method still achieve stabler (*i.e.*, smaller standard deviation) $\Delta\cos$ results on dSprites. In conclusion, our proposed method can accurately discover the biased attribute and is more disentangled w.r.t. the target attribute. For qualitative comparison, we randomly sample an experiment setting for each dataset and generate traversal images based on the predicted hyperplanes. As shown in Fig. 3, compared with other methods, our method can accurately discover the biased attribute and the traversal images of our method do not vary in terms of the target attribute. For example, in Fig. 3 (b) the traversal images of our method change by *shape* (*i.e.*, from "heart" to "ellipse"), which is the ground-truth biased attribute, while traversal images of baseline methods vary over non-biased attributes such as *ori-*

	$\mathcal{L}_H$	$\mathcal{L}_\perp$	$\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{b}\rangle\rvert \uparrow$	$\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{t}\rangle\rvert \downarrow$	$\Delta\cos \uparrow$
SmallNORB			0.25±0.15	0.27±0.18	-0.02±0.23
		✓	0.23±0.18	**0.10±0.11**	**0.12±0.21**
	✓		**0.27±0.16**	0.28±0.17	-0.01±0.25
	✓	✓	0.25±0.17	0.15±0.13	0.10±0.24
dSprites			0.20±0.13	0.21±0.13	-0.01±0.18
		✓	0.17±0.14	**0.13±0.11**	**0.05±0.18**
	✓		**0.21±0.13**	0.21±0.13	0.00±0.18
	✓	✓	**0.21±0.13**	0.19±0.13	0.01±0.18

Table 2: Ablation study on *orthogonalization penalty* ($\mathcal{L}_\perp$) and Hessian Penalty [44] ($\mathcal{L}_H$). ✓denotes the penalty is used. Note that all rows used $\mathcal{L}_V$. We incorporate $\mathcal{L}_H$ into our method. Although adding $\mathcal{L}_H$ helps to improve $\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{b}\rangle\rvert$, it seriously harms the $\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{t}\rangle\rvert$. Overall, our final method (second row in each dataset) performs the best in $\Delta\cos$.

	method	$\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{b}\rangle\rvert \uparrow$	$\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{t}\rangle\rvert \downarrow$	$\Delta\cos \uparrow$
CelebA	$\mathcal{L}_H$	0.02±0.02	0.02±0.0003	0.0005±0.02
	Ours	**0.06±0.01**	**0.002±0.001**	**0.06±0.01**
FFHQ	$\mathcal{L}_H$	0.05±0.01	0.01±0.008	0.03±0.004
	Ours	**0.17±0.11**	**0.002±0.002**	**0.17±0.11**

Table 3: Results on CelebA [34] and FFHQ [23] datasets. We omit %leading since our method leads in all experiment settings (*i.e.*, %leading (Ours) = 100 %).

entation or *position x* (*i.e.*, the object is rotating or moving horizontally).

Ablation Study on $\mathcal{L}_\perp$ and $\mathcal{L}_H$ We conduct ablation study on the *orthogonalization penalty* ($\mathcal{L}_\perp$). The results in Tab. 2 show that $\mathcal{L}_\perp$ is helpful in decreasing $\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{t}\rangle\rvert$, proving its effectiveness in being more orthogonal w.r.t. target attribute. However, $\mathcal{L}_\perp$ also decreases $\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{b}\rangle\rvert$. We suggest that $\mathcal{L}_\perp$ may make the optimization problem harder due to the additional constraint. Furthermore, the good results of Hessian Penalty (denoted as $\mathcal{L}_H$) in terms of $\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{b}\rangle\rvert$ motivates us to combine our method with Hessian Penalty via joint optimization. The results show that combining with $\mathcal{L}_H$ can achieve improvement in $\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{b}\rangle\rvert$, as well. However, it still harms the performance in $\lvert\cos\langle\hat{\mathbf{w}}_\mathsf{b}, \mathbf{w}_\mathsf{t}\rangle\rvert$, suggesting that $\mathcal{L}_H$ learns an averaged hyperplane between the biased and the target attributes. We also conduct **additional ablation studies** on the set of known attributes and the distribution of training data of generative models. Results are shown in Appx. C.

5.2. Experiment on Face Images

Experiment Settings We use CelebA [34] and FFHQ [23] datasets for discovering biased attributes of face images. CelebA is a dataset of face images of celebrities with 40 annotated attributes. FFHQ dataset contains 70,000 high-quality face images. We choose *gender* as the target attribute to train two ResNet-18 [19] networks as the target attribute classifiers supervised by the attribute labels on two datasets, respectively. For FFHQ dataset, we use the *gender* annotations from [42]. Note that different from the first experiment, we do *not* sample datasets with any skewed distributions because we want to discover the underlying biased attributes in the original face datasets. We choose two generative models: two StyleGAN [23] networks pretrained on CelebA-HQ [22] and FFHQ [23] datasets, respectively. We use the "style" latent space $\mathcal{W}$ of the StyleGAN, where attributes are more linearly separable than input noise latent space [23]. Hessian Penalty ($\mathcal{L}_H$) is used as the baseline method. Since CelebA and FFHQ are in-the-wild datasets, we do not know the real ground-truth biased attributes. Therefore, for the quantitative evaluation, we obtain the pseudo-ground-truth of the biased attribute (see Appx. A.4). Due to the absence of real ground-truth of the biased attribute, we do not use known attributes in *orthogonalization penalty* (*i.e.*, $K = \varnothing$).

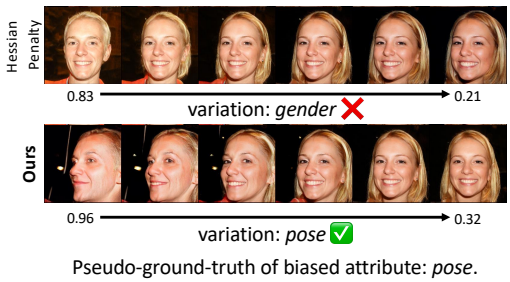

Figure 4: Qualitative comparison of the traversal images of predicted biased attribute synthesized by StyleGAN [23] pretrained on (a) FFHQ [23] dataset. The target attribute classifier is trained on FFHQ. The target attribute is *gender* and the pseudo-ground-truths of biased attribute is *pose*. The numbers under the images are *gender* classifier's predictions on whether the *gender* attribute's value is "male." Our method correctly discover the *pose* biased attribute.

Results Similar to the first experiment, we run experiments under all settings of (target attribute, biased attribute, generative model). Results averaged across all settings are reported in Tab. 3. The results of each experiment setting are in Appx. E. Our method beats the Hessian Penalty method in all metrics and in each experiment setting. The qualitative comparisons are shown in Fig. 4. The traversal images of the Hessian Penalty method vary in terms of the target attribute *gender* (*i.e.*, male to female). In contrast, our method correctly predicts the pseudo-ground-truth of the biased attribute: *pose*. More examples are shown in Appx. F.1.

Discovering Other Biased Attributes In this experiment, we try finding the biased attributes other than some known attributes. In the *orthogonalization penalty* ($\mathcal{L}_\perp$), we let the set of the known attribute K to be four attributes: *age*, *eyeglasses*, *pose*, and *smile*, whose hyperplanes are provided in [46]. Results in Fig. 5 show that our method can successfully discover other biased attributes such as *lighting* and *bald*, proving the effectiveness of the known attributes in $\mathcal{L}_\perp$. One

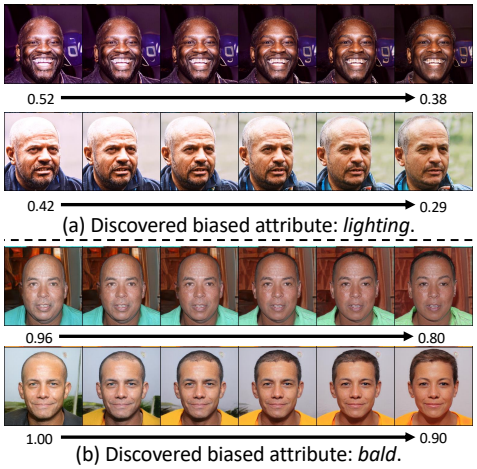

Figure 5: Discovered biased attributes by setting the set of known attributes K to all considered attributes generated by StyleGAN [23] pretrained on FFHQ [23] dataset. The target attribute classifiers in (a) and (b) are trained on CelebA and FFHQ, respectively. The numbers under the images are *gender* classifier's predictions on whether the *gender* attribute's value is "male."

may regard the variations that do not switch 0.5-threshold (*e.g.*, 1.00 to 0.90 in Fig. 5) are not meaningful. We believe such variation is still valuable and include a discussion in Appx. H.3. We also conduct a user study (see Appx. G) to invite subjects to name the attributes from more traversal images, whose results prove that our method finds biased attributes difficult for the Hessian Penalty method.

5.3. Experiment on Images from Other Domains

Finally, we apply our method on images from other domains, including object (*e.g.*, *cat*) and scenes (*e.g.*, *bedroom*) categories in LSUN dataset [55]. StyleGAN and StyleGAN2 pretrained on the images from each category are used as generative models and weights are obtained from [47]. Since each generator is trained on only one category of images (*e.g.*, the cat generator is only trained on cat images), we only use $\mathcal{L}_V$ and do not use $\mathcal{L}_\perp$ because the target attribute value (*i.e.*, object or scene category) is fixed for each generator. We choose ResNet-18 [19] pretrained on ImageNet as the object classifier and ResNet-18 pretrained on Places365 as the scene classifier. We show some biased attributed discovered by our method in Fig. 6. Our method successfully discovers biased attributes such as *shade of fur color*, *is Eiffel Tower*, *layout*, and *number of beds* in cat, tower, conference room, and bedroom classifiers, respectively, which could be hard for human to speculate in advance. We also conduct a user study (see Appx. G) on letting subjects name the biased attributes from more traversal images, which verifies our method discovers biased attributes that are difficult for Hessian Penalty. This proves the generalizability of our method for discovering biased attributes in various image domains.

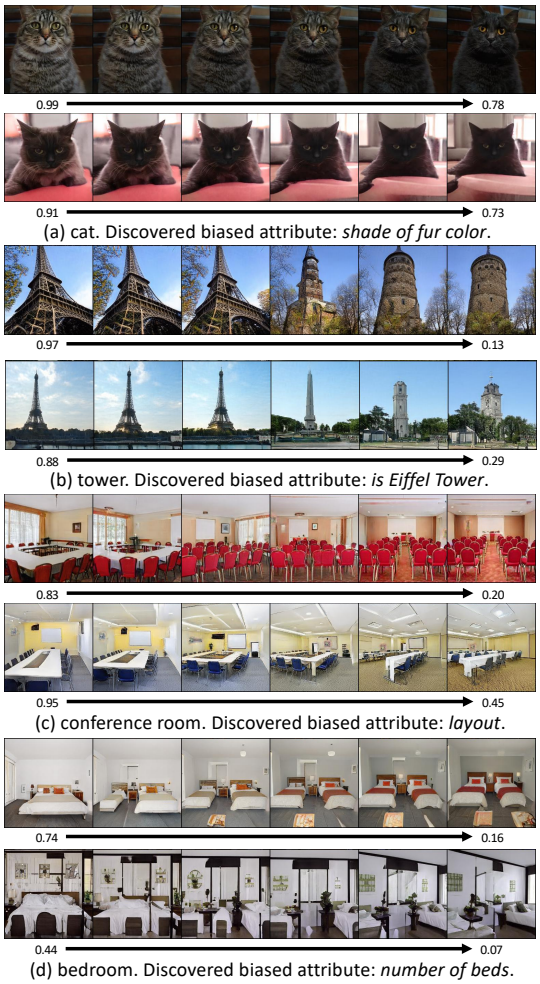

Figure 6: Discovered biased attribute of classifiers for classifying *cat*, *tower*, *conference room*, *bedroom* images. Numbers below images are predicted probability by the classifier.

6. Conclusion

In this work, we propose a new problem for finding the unknown biased attribute of a classifier without presumptions or labels. To tackle this new problem, a novel method is proposed for this task by optimizing *total variation loss* and *orthogonalization penalty*. The comprehensive experiments prove that our method is effective and can discover biased attributes in multiple domains. In the appendix, we discuss the limitations, future directions, and the related methods and areas that can be benefited from this new *unknown biased attribute discovery task*.

Acknowledgements. This work has been partially supported by the National Science Foundation (NSF) under Grants 1764415, 1813709, 1909912, and 1934962. The article solely reflects the opinions and conclusions of its authors but not the funding agents.

References

[1] Amir H. Abdi, Purang Abolmaesumi, and Sidney Fels. Variational Learning with Disentanglement-PyTorch. In *Advances in Neural Information Processing Systems Workshop*, 2019. 12

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual Question Answering. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015. 3

[3] Guha Balakrishnan, Yuanjun Xiong, Wei Xia, and Pietro Perona. Towards causal benchmarking of bias in face analysis algorithms. In *The European Conference on Computer Vision (ECCV)*, 2020. 1, 3, 11

[4] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3D faces. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, 1999. 2

[5] Joy Buolamwini and Timnit Gebru. Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification. In *ACM Conference on Fairness, Accountability, and Transparency*, 2018. 1, 2

[6] Ricky T. Q. Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating Sources of Disentanglement in Variational Autoencoders. In *Advances in Neural Information Processing Systems*, 2018. 3, 6

[7] Jinwoo Choi, Chen Gao, Joseph C E Messou, and Jia-Bin Huang. Why Can't I Dance in the Mall? Learning to Mitigate Scene Bias in Action Recognition. In *Advances in Neural Information Processing Systems*, 2019. 1, 2

[8] Kristy Choi, Aditya Grover, Trisha Singh, Rui Shu, and Stefano Ermon. Fair Generative Modeling via Weak Supervision. In *International Conference on Machine Learning*, 2020. 3

[9] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, (3), 1995. 11

[10] Elliot Creager, David Madras, Joern-Henrik Jacobsen, Marissa Weis, Kevin Swersky, Toniann Pitassi, and Richard Zemel. Flexibly Fair Representation Learning by Disentanglement. In *International Conference on Machine Learning*, 2019. 2, 3

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 2

[12] Emily Denton, Ben Hutchinson, Margaret Mitchell, Timnit Gebru, and Andrew Zaldivar. Detecting Bias with Generative Counterfactual Face Attribute Augmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019. 2, 3, 21

[13] Emily Denton, Ben Hutchinson, Margaret Mitchell, Timnit Gebru, and Andrew Zaldivar. Image Counterfactual Sensitivity Analysis for Detecting Unintended Bias. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019. 2, 3, 21

[14] Burkay Donderici, Caleb New, and Chenliang Xu. Assembling Semantically-Disentangled Representations for Predictive-Generative Models via Adaptation from Synthetic Domain. *arXiv:2002.09818 [cs, eess]*, 2020. 3

[15] Emilien Dupont. Learning Disentangled Joint Continuous and Discrete Representations. In *Advances in Neural Information Processing Systems*, 2018. 3

[16] 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*, 2014. 4

[17] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. In *International Conference on Learning Representations*, 2015. 21

[18] Moritz Hardt, Eric Price, Eric Price, and Nati Srebro. Equality of Opportunity in Supervised Learning. In *Advances in Neural Information Processing Systems*, 2016. 2

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 7, 8

[20] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. Beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *International Conference on Learning Representations*, 2017. 2, 3, 6, 11, 12, 13, 14

[21] Jungseock Joo and Kimmo Kärkkäinen. Gender Slopes: Counterfactual Fairness for Computer Vision Models by Attribute Manipulation. In *Proceedings of the 2nd International Workshop on Fairness, Accountability, Transparency and Ethics in Multimedia*, 2020. 3, 21

[22] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive Growing of GANs for Improved Quality, Stability, and Variation. In *International Conference on Learning Representations*, 2018. 2, 7, 12

[23] Tero Karras, Samuli Laine, and Timo Aila. A Style-Based Generator Architecture for Generative Adversarial Networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 7, 8, 11, 12, 15, 17, 18, 21

[24] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and Improving the Image Quality of StyleGAN. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3, 12

[25] Hyunjik Kim and Andriy Mnih. Disentangling by Factorising. In *International Conference on Machine Learning*, 2018. 3

[26] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations*, 2015. 11

[27] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014. 3, 4, 6, 12

[28] Adam Kortylewski, Bernhard Egger, Andreas Schneider, Thomas Gerig, Andreas Morel-Forster, and Thomas Vetter. Empirically Analyzing the Effect of Dataset Biases on Deep Face Recognition Systems. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2018. 2

[29] Adam Kortylewski, Bernhard Egger, Andreas Schneider, Thomas Gerig, Andreas Morel-Forster, and Thomas Vetter. Analyzing and Reducing the Damage of Dataset Bias to Face Recognition With Synthetic Data. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2019. 2

[30] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational Inference of Disentangled Latent Concepts from Unlabeled Observations. In *International Conference on Learning Representations*, 2018. 3, 6, 12

[31] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual Fairness. In *Advances in Neural Information Processing Systems*, 2017. 2, 3, 21

[32] Yann LeCun, Fu Jie Huang, and Leon Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004. 2, 5, 6, 11, 13, 14, 15, 16, 17

[33] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *The European Conference on Computer Vision (ECCV)*, 2014. 21

[34] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep Learning Face Attributes in the Wild. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015. 2, 7, 12, 15

[35] Francesco Locatello, Gabriele Abbati, Thomas Rainforth, Stefan Bauer, Bernhard Schölkopf, and Olivier Bachem. On the Fairness of Disentangled Representations. In *Advances in Neural Information Processing Systems*, 2019. 3

[36] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging Common Assumptions in the Unsupervised Learning of Disentangled Representations. In *International Conference on Machine Learning*, 2019. 6, 12, 21

[37] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*, 2018. 21

[38] Varun Manjunatha, Nirat Saini, and Larry S. Davis. Explicit Bias Discovery in Visual Question Answering Models. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3

[39] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model Cards for Model Reporting. In *ACM Conference on Fairness, Accountability, and Transparency*, 2019. 1

[40] Vidya Muthukumar, Tejaswini Pedapati, Nalini Ratha, Prasanna Sattigeri, Chai-Wah Wu, Brian Kingsbury, Abhishek Kumar, Samuel Thomas, Aleksandra Mojsilovic, and Kush R. Varshney. Understanding Unequal Gender Classification Accuracy from Face Images. *arXiv:1812.00099 [cs, stat]*, 2018. 2

[41] J. A. Nelder and R. W. M. Wedderburn. Generalized Linear Models. *Journal of the Royal Statistical Society: Series A (General)*, (3), 1972. 11

[42] Roy Or-El, Soumyadip Sengupta, Ohad Fried, Eli Shechtman, and Ira Kemelmacher-Shlizerman. Lifespan Age Transformation Synthesis. In *The European Conference on Computer Vision (ECCV)*, 2020. 7

[43] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*, 2019. 11

[44] William Peebles, John Peebles, Jun-Yan Zhu, Alexei Efros, and Antonio Torralba. The Hessian Penalty: A Weak Prior for Unsupervised Disentanglement. In *The European Conference on Computer Vision (ECCV)*, 2020. 3, 6, 7, 11, 14, 16

[45] Mhd Hasan Sarhan, Nassir Navab, and Shadi Albarqouni. Fairness by Learning Orthogonal Disentangled Representations. In *The European Conference on Computer Vision (ECCV)*, 2020. 2, 3

[46] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the Latent Space of GANs for Semantic Face Editing. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 4, 7, 11, 12

[47] Yujun Shen, Yinghao Xu, Ceyuan Yang, Jiapeng Zhu, and Bolei Zhou. GenForce. 2020. 8, 12

[48] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. InterFaceGAN: Interpreting the Disentangled Face Representation Learned by GANs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2, 4

[49] Antonio Torralba and Alexei A. Efros. Unbiased look at dataset bias. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 21

[50] Matthew J. Vowels, Necati Cihan Camgoz, and Richard Bowden. NestedVAE: Isolating Common Factors via Weak Supervision. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3

[51] Andrey Voynov and Artem Babenko. Unsupervised Discovery of Interpretable Directions in the GAN Latent Space. In *International Conference on Machine Learning*, 2020. 3

[52] Angelina Wang, Arvind Narayanan, and Olga Russakovsky. REVISE: A Tool for Measuring and Mitigating Bias in Image Datasets. In *The European Conference on Computer Vision (ECCV)*, 2020. 3

[53] 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 *The IEEE International Conference on Computer Vision (ICCV)*, 2019. 1

[54] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 3, 13, 21

[55] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. *arXiv:1506.03365 [cs]*, 2016. 8, 12

[56] 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*, (6), 2018. 2