Supplementary Material for
A Conservative Approach for Unbiased Learning on Unknown Biases

Myeongho Jeon[1], Daekyung Kim[2], Woochul Lee[1], Myungjoo Kang[1], Joonseok Lee[1]
[1]Seoul National University, [2]Monitor Corporation
{andyjeon, sunnmoon137, woochulee, mkang, joonseok}@snu.ac.kr

1. Biased CelebA-HQ

As argued in the paper, even though hair length is not biologically correlated to gender, the machine learning algorithm could perceive it as a decisive factor for gender discrimination and give over-credence on that attribute. To quantitatively estimate the degree of such a bias, we present a new attribute 'hair length' for CelebA-HQ dataset [7], and constitute the Biased CelebA-HQ based on this attribute. We design two biased datasets, the 'extreme bias 1 (EB1)' only with the common combinations, (female, long hair) and (male, short hair), and 'extreme bias 2 (EB2)' with the other combinations, (female, short hair) and (male, long hair). The sample images of EB2 are exhibited in Fig. 1. The biased CelebA-HQ is constructed as follows:

1. The annotation in the CelebA-HQ is used to split the female and male images. After dividing the images by gender, we manually inspected them to remove incorrectly labeled samples. There are a small minority of incorrect annotations, but it could give an irresistible effect on unbiased modeling.

2. We manually labeled all images with hair length to {short(0), long(1)}. Images with intermediate length of hair were excluded (See examples in Fig. 2A).

3. For all the images, we take off exceptional cases that are tricky to be categorized (See Fig. 2B,C,D,E).

4. As a result, we got 15,441 images for (female, long), 8,986 for (male, short), 309 for (female, short), and 565 images for (male, long). In total, we have 25,301 images in the dataset.

5. We union (female, long hair) and (male, short hair) images to the first set, namely 'extreme bias 1 (EB1)'. The rests, (female, short hair) and (male, long hair), create the other set, 'extreme bias 2 (EB2)'. Then, we sample 1,000 images for each pair from EB1 and name it val-EB1, leaving the remainders as train-EB1.

6. We realize that the number of EB2 images is insufficient, so supplement them from CelebA dataset [8]. As a result, 3,191 (female, short hair) samples and 1,335 (male, long hair) samples are additionally included. We manually removed duplicate images in EB2, since the CelebA-HQ is the subset of the CelebA.

7. Among these EB2 images, we select 1,000 images for each pair for testing and 0.5% of each male and female for EB1 are sampled to make the 'utmost bias 1 (UB1)'. During sampling, we give priority to the images from CelebA-HQ, then consider CelebA images. In consequence, 72 (female, short hair) images and 38 (male, long hair) images are added to train-EB1 resulting in UB1.

The resulting number of images in training and test dataset is summarized in Tab. 1. Although most of EB2 images from CelebA are not utilized for this experiment, we provide all the annotations to contribute to future research.

	EB1		EB2	
Gender	female	male	female	male
Hair	long	short	short	long
train	14,441	7,986	72	38
val	1,000	1,000	1,000	1,000

Table 1. **Train and validation set of the Biased CelebA-HQ.**

2. Exploration on the Data Distribution

In this section, we exhibit the details of the training-validation data distribution for CelebA-HQ and UTK-Face [11].

CelebA-HQ. The UB1, which is used for training, contains all 4 combinations in the training set, in total 22,537 images. The val-EB1 consists of 1,000 images for each common combination, (female, long hair) and (male, short hair) from the validation partition, while the val-EB2 consists of 1,000 images from the other combinations, (female,

Figure 1. **Examples of Extreme bias 2 (EB2) dataset**. The image samples of the (female, short hair) and (male, long hair) pair. The EB1 image samples are shown in the main text.

Figure 2. **Exceptional cases**. The images above were excluded from our experiments for the following reasons. A: hair with intermediate length. B: hair is obscured by something such as a hat. C: the gender is cannot be recognized. D: the hair is guessed to be long, but it is not clearly shown. E: the hair is covered by the image boundary.

short hair) and (male, long hair), thus each set consists of 2,000 images in total. Figure 3 shows the overall data distribution of the Biased CelebA-HQ.

UTKFace. Between 'age', 'gender', and 'skin tone' annotations provided in UTKFace, age is not used since the annotation pairs for age ({age, gender}, {age, skin tone}) are imbalanced. In Fig. 4, the first graph shows (female, old) images are much fewer than (female, young) ones, and the second graph shows (dark skin, old) images are much fewer than (bright skin, young). This skewed distribution is not appropriate for unbiased modeling experiment. If we predict age with gender bias and α is 0.2, for example, the number of (female, young) images is similar to that of (female, old) and hence the dataset becomes unbiased (See the first graph in Fig. 4). Thus, only with the {gender, skin tone} labels, we perform 1) skin tone prediction with gender bias and 2) gender prediction with skin tone bias.

For the skin tone prediction with gender bias, we split the images into extremely biased sets. Then, we add EB2 image samples (with up to $\alpha = 0.2$) to train-EB1, and name it UB1. UB2 is created in a similar manner. The unbiased test set, composed of 300 images for each pair of {gender, skin tone}, thereby 1,200 images in total. The gender prediction with skin tone bias scenario is performed in the same way. Fig. 5 shows the overall data distribution of UB1, UB2, and the unbiased test sets.

3. Training Details

In this section, we report training details that are not specified in the paper. The details for each dataset are as

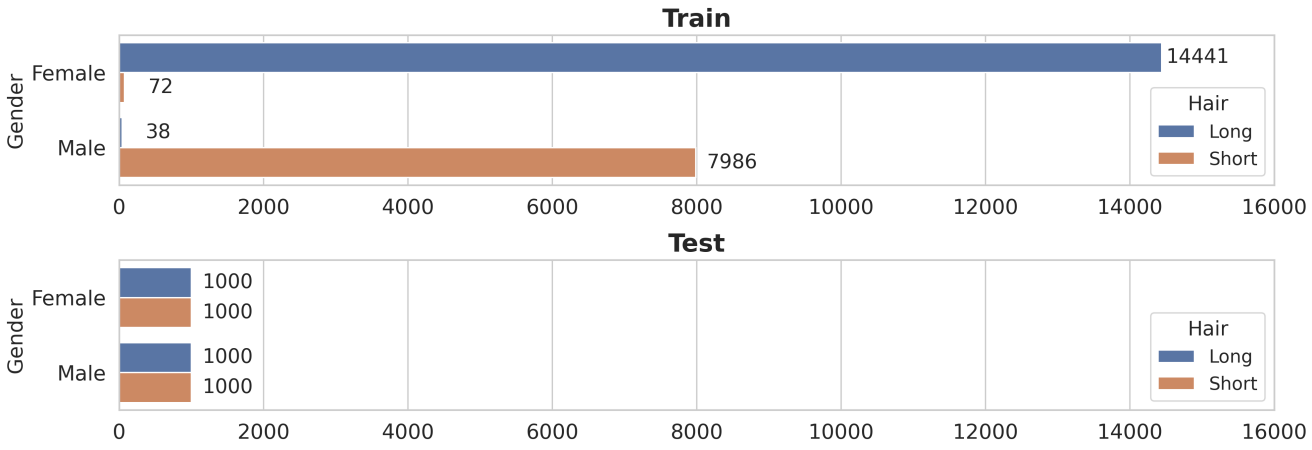

Figure 3. **The overall data distribution of biased CelebA-HQ.**

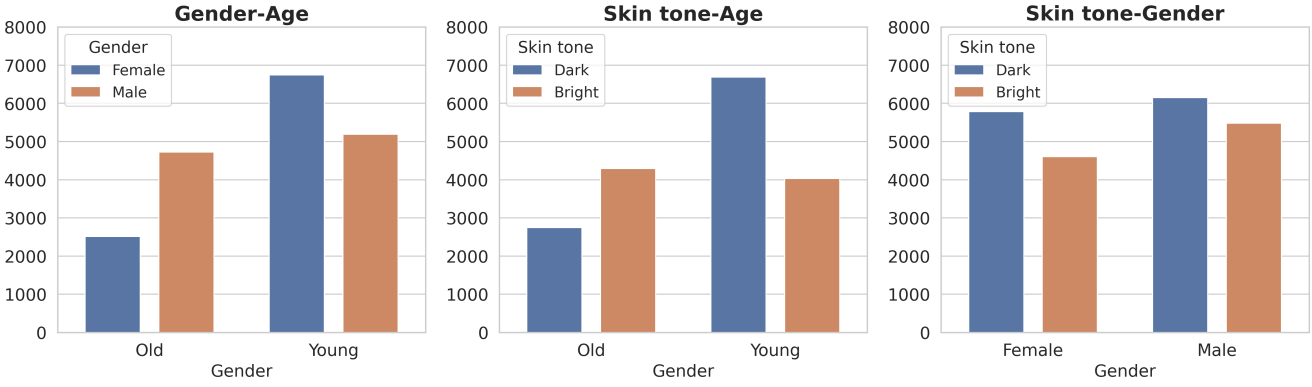

Figure 4. **The data distribution of each annotation pair for UTKFace.**

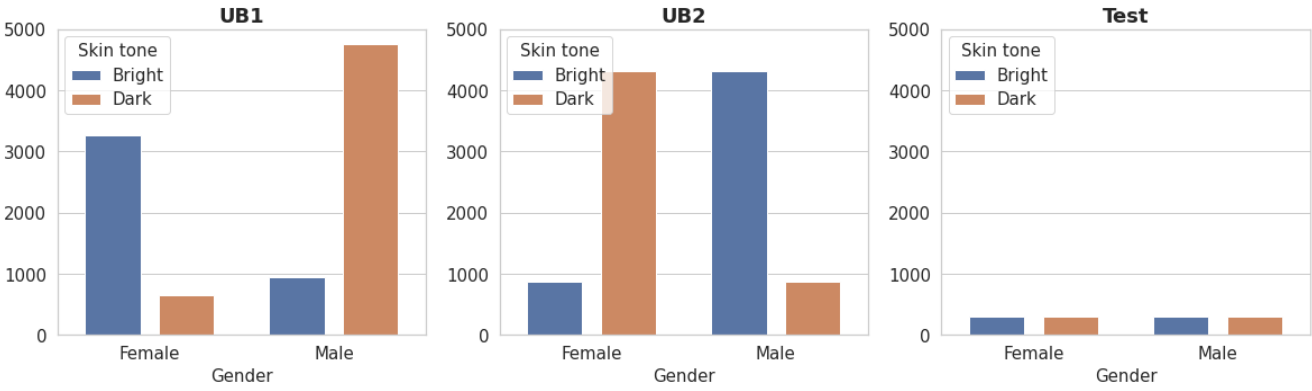

Figure 5. **The overall data distribution of UTKFace.**

follows:

In all the experiments, We set $L = 3$ for AlexNet and $L = 5$ for VGG11, ResNet18. In addition, we set $W = H = 7, C = 64$ for g_l.

CelebA-HQ. The input images are resized to (224, 224) and scaled to $0 - 1$. To train the base model that acts as the feature extractor in our method, ImageNet-pretrained weights are utilized for initialization. During training, we set the batch size as 32 and use the Adam optimizer [5] with weight decay (0.0005), $\beta 1$ (0.9), $\beta 2$ (0.999), and eps

(10^{-8}). The AMSGrad variant of the model is not used in our optimizing procedure. We train our model for 20 epochs, the first 10 epochs with learning rate 10^{-4}, then with 10^{-5} for the last 10 epochs. We set λ of orthogonal loss as 10.

UBnet is initialized with Xavier [2]. The input image and the other experimental settings for training UBnet are exactly the same as the base model.

UTKFace. The input images are resized to (224, 224) and standardized. The base model is initialized by ImageNet-pretrained weights. We use the batch size 512 and train the model for 20 epochs by AdamP optimizer [4] with weight decay (0.0005), $\beta 1$ (0.9), $\beta 2$ (0.999), and eps (10^{-8}). The learning rate is initially set to 10^{-3} and is decayed by factor 0.1 for every 10 epochs. We set λ of orthogonal loss as 1.

UBnet is initialized with Xavier. All the other setups for UBnet are the same as that of the base model.

ImageNet. The input images are resized to (224, 224) and standardized. To train the base model, the weights are initialized with Xavier. The learning rate is initially set to 10^{-4} and is decayed by cosine annealing to be 0 at the maximum epochs of 120. We use the batch size 512 and train UBnet for 120 epochs via Adam optimizer with weight decay (0.0005), $\beta 1$ (0.9), $\beta 2$ (0.999), and eps (10^{-8}). We set λ of orthogonal loss as 10.

The input image and the experimental settings for training UBnet are exactly the same as base model.

4. Evaluations on other Base Models

The contribution of the base model depends on the experiment settings such as task, dataset, and applied subcomponents. Therefore, most works search for the most appropriate one according to their modeling purpose. The base models for HEX [10] and Rebias [1] are AlexNet [6] and ResNet18 [3], respectively. Although the results with VGG11 [9] are reported in the paper, we evaluate the performance of our UBnet on AlexNet and ResNet18 to estimate if the proposed method generalizes better than competing models on the same base model condition. From the experiments in Tab. 2, UBnet(alex) and UBnet(res) outperform HEX and Rebias, respectively.

4.1. Ablation Study on Model Size.

We explore the effect of the model size in terms of the number of parameters. As seen in Tab. 3, our model significantly outperforms the base models even with a smaller number of parameters (in the case of VGG16) on EB2 and test set. This result shows that the increased model size (by 0.2%) is not the main factor of the improved performance.

Method	Acc(EB1)	Acc(EB2)	Acc(Test)
Base model	99.38(±0.31)	51.22(±1.73)	75.30(±0.93)
HEX	92.50(±0.67)	50.85(±0.37)	71.68(±0.50)
Rebias	99.05(±0.13)	55.57(±1.43)	77.31(±0.71)
UBnet(alex)	99.40(±0.26)	51.60(±0.36)	75.50(±0.26)
UBnet(res)	**99.57(±0.12)**	57.48(±2.23)	78.53(±1.16)
UBnet(vgg)	99.18(±0.18)	**58.22(±0.64)**	**78.70(±0.24)**

Table 2. **Results on CelebA-HQ.** Acc(EB1), Acc(EB2) and Acc(test) denote accuracy on val-EB1, val-EB2, and Test sets respectively for the model trained on UB1.

Method Params	VGG11 128,774K	VGG13 128,959K	VGG16 134,268K	UBnet 129,045K
Acc(EB1)	99.38	**99.58**	99.42	99.18
Acc(EB2)	51.22	52.45	54.50	**58.22**
Acc(test)	75.30	76.02	76.96	**78.70**

Table 3. **Ablation Study on Model Size.** The proposed method shows the best performance on Acc(EB2) and Acc(test).

References

[1] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *Proc. of the International Conference on Machine Learning (ICML)*, 2020. 4

[2] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proc. of the international conference on artificial intelligence and statistics (AISTATS)*, 2010. 4

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2016. 4

[4] Byeongho Heo, Sanghyuk Chun, Seong Joon Oh, Dongyoon Han, Sangdoo Yun, Gyuwan Kim, Youngjung Uh, and Jung-Woo Ha. AdamP: Slowing down the slowdown for momentum optimizers on scale-invariant weights. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2021. 4

[5] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2015. 3

[6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in neural information processing systems (NIPS)*, 2012. 4

[7] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. MaskGAN: Towards diverse and interactive facial image manipulation. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[8] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proc. of International Conference on Computer Vision (ICCV)*, 2015. 1

[9] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2015. 4

[10] Haohan Wang, Zexue He, Zachary C Lipton, and Eric P Xing. Learning robust representations by projecting superficial statistics out. In *Proc. of the International Conference on Learning Representations (ICLR)*, 2019. 4

[11] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. In *Proc. of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2017. 1