Supplementary Material: Fair Visual Recognition in Limited Data Regime using Self-Supervision and Self-Distillation

1. Effect of Biased Data on Model Predictions

1.1. Terminologies used

1. grayscale CIFAR-10S - All images in CIFAR-10S converted to grayscale.
2. color CIFAR-10S - All images in CIFAR-10S replaced by the corresponding original color images in CIFAR-10.
3. grayscale L-CIFAR-10S - All images in L-CIFAR-10S converted to grayscale.
4. color L-CIFAR-10S - All images in L-CIFAR-10S replaced by the corresponding original color images in CIFAR-10.
5. GrayTrainColorTest (GTCT) on CIFAR-10S - Model trained on grayscale CIFAR-10S and evaluated on a test set containing only color images.
6. GrayTrainColorTest (GTCT) on L-CIFAR-10S - Model trained on grayscale L-CIFAR-10S and evaluated on a test set containing only color images.
7. ColorTrainColorTest (CTCT) on CIFAR-10S - Model trained on color CIFAR-10S and evaluated on a test set containing only color images.
8. ColorTrainColorTest (CTCT) on L-CIFAR-10S - Model trained on color L-CIFAR-10S and evaluated on a test set containing only color images.

1.2. Effect of Data Bias with Sufficient Data

The authors in [5] experimentally show that a baseline ResNet-18 [2] model trained on CIFAR-10S achieves $89.0 \pm 0.5\%$ classification accuracy on the color test set. Whereas the same model trained on grayscale CIFAR-10S (GTCT) achieves 93.0% classification accuracy on the color test set [5], which is a significant increase. The authors attribute this problem to the skewed data in the CIFAR-10S dataset. As a result, the model is biased towards predicting one among the five color domain classes in the case of colored images since these classes contain mostly color images. The authors in [5] propose that the goal of any bias mitigating approach should be to achieve a classification accuracy at least close to that of GTCT on CIFAR-10S (93.0%). It should also ideally approach the classification accuracy of CTCT on CIFAR-10S which is 95.0%. The performance gap in the baseline (grayscale CIFAR-10S vs. CIFAR-10S) due to the bias is 4%. Our approach trained on sufficient data (CIFAR-10S) achieves a classification accuracy of 91.1% on the color test, which is significantly higher than the baseline and reduces this gap from 4% to around 2%. This also validates that generalization and representation learning matter even in the case of fair visual recognition with sufficient data.

The authors in [5] compare various bias mitigation approaches on the CIFAR-10S dataset. The experimental results indicate that the domain independent strategy proposed in [5] achieves the highest accuracy of 92.4% on the color test set as compared to the other approaches. The domain discriminative training approach achieves a classification accuracy of 91.2% on the color test set. These performances are close to the 93% classification accuracy achieved by GTCT on CIFAR-10S. Therefore, the gap in the performance of the domain discriminative and domain independent training approaches compared to GTCT on the CIFAR-10S dataset are 1.8% and 0.6%, respectively. Our approach augmented with the domain independent method trained on CIFAR-10S achieves a classification accuracy of 93.2% on the color test, which outperforms the domain independent (92.4%) model and also slightly outperforms GTCT (93%) on CIFAR-10S.

2. Experiments

2.1. Datasets

Introductory details regarding the datasets have been provided in Sec. 5.1 in the main paper. All the CIFAR-10S variants retain the 10 classes and 50,000 training images of the CIFAR-10 [4] dataset but modify the training images to introduce a bias using a domain attribute containing two domains, d1 and d2. The d1 domain has five classes, each containing 95% d1 type images and 5% d2 type images (95-5% skew). The remaining five classes belong to the d2 do-

main and contain 95% d2 type and 5% d1 type images each (95-5% skew). Further details are given below.

1. CIFAR-10S dataset: The domain attribute in this dataset is whether the image is colored (standard CIFAR-10 color images) or grayscale (grayscale version of CIFAR-10 images). The color domain has five classes, each containing 95% color images and 5% grayscale images (95-5% skew). The remaining five classes belong to the grayscale domain and contain 95% grayscale and 5% color images each (95-5% skew).

2. CIFAR-10S-i: The domain attribute in this dataset is whether the image is a standard CIFAR-10 color image or a downsized image (32×32) of the same class from the ImageNet dataset. The CIFAR-10 color domain has five classes, each containing 95% color images and 5% ImageNet images (95-5% skew). The remaining five classes belong to the ImageNet domain and contain 95% ImageNet images and 5% CIFAR-10 color images each (95-5% skew).

3. CIFAR-10S-c28: The domain attribute in this dataset is whether the image is a standard CIFAR-10 color image of size 32×32 or a CIFAR-10 image cropped to 28×28 from the center and resized/upsampled to 32×32 (c28). The CIFAR-10 color domain has five classes, each containing 95% color images and 5% c28 images (95-5% skew). The remaining five classes belong to the c28 domain and contain 95% c28 images and 5% CIFAR-10 color images each (95-5% skew).

4. CIFAR-10S-d16: The domain attribute in this dataset is whether the image is a standard CIFAR-10 color image of size 32×32 or a CIFAR-10 image downsized to 16×16 and resized/upsampled to 32×32 (d16). The CIFAR-10 color domain has five classes, each containing 95% color images and 5% d16 images (95-5% skew). The remaining five classes belong to the d16 domain and contain 95% d16 images and 5% CIFAR-10 color images each (95-5% skew).

5. CIFAR-10S-d8: The domain attribute in this dataset is whether the image is a standard CIFAR-10 color image of size 32×32 or a CIFAR-10 image downsized to 8×8 and resized/upsampled to 32×32 (d8). The CIFAR-10 color domain has five classes, each containing 95% color images and 5% d8 images (95-5% skew). The remaining five classes belong to the d8 domain and contain 95% d8 images and 5% CIFAR-10 color images each (95-5% skew).

We have provided concise details regarding the CIFAR-10S, CIFAR-10S-i, CIFAR-10S-c28, CIFAR-10S-d16, and CIFAR-10S-d8 datasets. For further details regarding these datasets, please refer to [5].

The L-CIFAR-10S, L-CIFAR-10S-i, L-CIFAR-10S-c28, L-CIFAR-10S-d16, and L-CIFAR-10S-d8 datasets are limited data versions of CIFAR-10S, CIFAR-10S-i, CIFAR-10S-c28, CIFAR-10S-d16, and CIFAR-10S-d8 datasets, respectively. They contain 5% of the images of their respective parent dataset but with the same skew level (95-5%). Specifically, we choose the first 5% images from each class while maintaining the same level of skew, i.e., we separately choose the first 5% images from each domain for every class. For example, in the case of L-CIFAR-10S, we separately choose 5% images from the color and grayscale images in each class. As a result, if a class contained 95% color images and 5% grayscale images or vice-versa, the ratio of color and grayscale images will still remain the same in every class of L-CIFAR-10S.

2.2. Compared Approaches

Introductory details regarding the compared approaches have been provided in Sec. 5.2 in the main paper. We provide further details regarding the compared approaches below:

1. Strategic sampling: In this approach, the rare (minority) images are strategically re-sampled to artificially balance the dataset in terms of the number of images of the two types/domains in each class. However, it also increases the chances of overfitting due to seeing the same images multiple times. It also increases the training time without providing any additional information [5].

2. Adversarial training approach: In this approach, a minimax objective is set to minimize the possibility that the protected attribute can be predicted using the features from the network while maximizing the classification power of the network. It is based on the idea that if the model cannot encode the information regarding the protected attribute, it will not be affected by the bias due to that attribute. In our paper, we perform experiments for both the techniques used in [5] under the adversarial approach. Specifically, we perform adversarial training using the uniform confusion loss $-(1/|D|)\sum_d \log q_d$ approach, and the loss reversal $\sum_d \mathbb{1}[\widehat{d} = d] \log q_d$ with gradient projection approach used in [5].

3. Domain discriminative training approach: In this approach, the protected attribute is explicitly modeled as opposed to the adversarial approach, and the correlation between the classes and the protected attribute is then explicitly removed during inference. The authors in [5] employ a simple approach of using an

ND-way classifier where N is the number of image classes, and D is the number of domains. They propose three types of inference for this approach. First approach, involves directly adding the probabilities for all the domains per class ($\arg\max_y \sum_d \mathrm{P}_{\mathrm{tr}}(y, d|x)$). But this does not take into account the prior information regarding the correlation between the classes and domains. The next two approaches involve first applying a prior shift based on this correlation ($\mathrm{P}_{\mathrm{te}}(y, d|x) = \mathrm{P}_{\mathrm{tr}}(y, d|x)/\mathrm{P}_{\mathrm{tr}}(y, d)$) and then either adding the probabilities for all the domains per class ($\arg\max_y \sum_d \mathrm{P}_{\mathrm{te}}(y, d|x)$) or taking the highest probability without adding the domain wise class probabilities ($\arg\max_y \max_d \mathrm{P}_{\mathrm{te}}(y, d|x)$). The authors in [5] also use Reducing Bias Amplification (RBA) [6] as an inference method.

4. Domain independent training approach: In this approach, the authors in [5] try to avoid the problems in the domain discriminative approach. One such problem is that the domain discriminative approach leads to learning decision boundaries among different domains in the same class, which may be unnecessary, especially in cases when the class prediction is already good. Therefore, in the domain independent training approach, separate classifiers are trained per domain but with a shared feature extraction network. The authors in [5] experiment with two inference methods: a) $\hat{y} = \arg\max_y \mathrm{P}_{\mathrm{te}}(y|d^*, x)$, if the domain d^* of the test image is known, b) $\hat{y} = \arg\max_y \sum_d \mathrm{s}(y, d, x)$, which is basically the sum of the classification layer activations for each domain per class.

We have provided concise details regarding the bias mitigation methods. For a more detailed discussion regarding these methods, refer to [5].

2.3. Implementation Details

For any given architecture (ResNet-18 or ResNet-50), we take the output of the last convolutional layer before the fully connected classification layer and use it for the SimSiam objective function. We add a multi-layer perceptron projection head and a multi-layer perceptron based prediction head as required by SimSiam. The multi-layer perceptron projection head has 3 fully connected layers, each having an output size of 2048 and followed by a batch normalization layer and ReLU activation. The multi-layer perceptron prediction head has 2 fully connected layers having output sizes of 512 and 2048, respectively. The first layer is followed by a batch normalization layer and ReLU activation. The training settings for SimSiam are the same as proposed in [1]. For self-distillation, we use $\kappa = 4$. For a fair comparison, we use the same training settings for all the methods as described in [5]. For the experiments involving the L-CIFAR-10S dataset and its variants, we train the model from scratch for 200 epochs using an SGD optimizer with an initial learning rate of $1\mathrm{e}-1$, weight decay of $5\mathrm{e}{-4}$, and momentum of 0.9. The learning rate is decreased by a factor of 10 after every 50 epochs. The training images are padded with 4 pixels, randomly flipped horizontally, and randomly cropped to 32×32 [5].

In the ResNet-50 backbone used for the L-CelebA experiments, the fully connected layer is replaced with 2 fully connected layers [5] with a dropout and a ReLU activation layer between them. We train the network using the binary cross-entropy loss for 50 epochs with a batch size of 32. We use the Adam optimizer [3] with a learning rate of $1\mathrm{e}{-4}$. For the L-CelebA experiments, we use a weighted mAP proposed in [5] to remove the gender bias in the test set. If an attribute is more prevalent among the women images, $BA = P_w/(P_m + P_w) - N_w/(N_m + N_w)$ where P_w, P_m refer to the number of images of women and men predicted to have this attribute. N_w, N_m refer to the actual number of women and men images in the training data. If an attribute is more prevalent among the men, $BA = P_m/(P_m + P_w) - N_m/(N_m + N_w)$. Since the objective of bias mitigation is to reduce the level of bias that was learned from the training data and the bias amplification score is expected to be negative as the model becomes fairer across genders [5].

In our approach, given an image, we apply random augmentations to create two different views. We feed them to the backbone network and perform the standard cross-entropy loss based training for image classification (or binary cross-entropy loss based training for multi-label classification). We also feed the features for both the views to the multi-layer perceptron projection head and then the prediction head in order to apply the SimSiam loss function as described in Eqs. 1, 2, 3, 4 in the main paper. In order to apply self-distillation, we minimize KL divergence between the logits/soft predictions of the student and teacher networks. For all the experiments, the domain labels are assumed to be available during training time.

We use the same experimental settings and metrics for comparing bias mitigation approaches as used in [5]. Please refer to [5] for further details.

References

[1] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2020.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016.

[3] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[4] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[5] Zeyu Wang, Klint Qinami, Ioannis Christos Karakozis, Kyle Genova, Prem Nair, Kenji Hata, and Olga Russakovsky. Towards fairness in visual recognition: Effective strategies for bias mitigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8919–8928, 2020.

[6] 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 *EMNLP*, 2017.