Supplementary for "CosFairNet:A Parameter-Space based Approach for Bias Free Learning"

Rajeev Ranjan Dwivedi[1]
rajeev22@iiserb.ac.in
Priyadarshini Kumari[2]
priyadarshini.kumari@sony.com
Vinod K Kurmi[1]
vinodkk@iiserb.ac.in

[1] Department of Data Science and Engineering
Indian Institute of Science Education and Research Bhopal
India
[2] Sony AI, USA

Abstract

In this supplementary material, we provide additional results on the Biased Action Recognition (BAR) dataset and more details about the other datasets used. We include results on the effect of the hyperparameter λ_c on model performance. Additionally, we examine the performance of our model on bias-aligned and bias-conflicting samples separately. We explore the impact of layer selection on model performance and characterize the effects of applying similarity versus dissimilarity. Furthermore, we provide Grad-CAM visualizations comparing our model to the vanilla model. Finally, we include a detailed algorithm of our model architecture.

1 Dataset Description

We assess our debiasing algorithm on four standard datasets. (1) The **Colored MNIST** dataset is the modified version of the MNIST dataset [8] where each digit is deliberately perturbed by a specific color to form a spurious correlation between bias attributes and target labels. Instead of creating our own version of the dataset, we use the Colored MNIST dataset as proposed in [11] to ensure a fair and direct comparison. This is because the dataset's construction significantly varies due to the changing coloring protocol. (2) The **Corrupted CIFAR-**10^1 [3] dataset introduces bias attributes through artificially generated distortions, such as fog, adjustments in brightness, or variations in saturation, which are synthetically applied to each class. (3) The **BFFHQ dataset** [5] is a gender-biased derivative of the widely-used Flickr-Faces-HQ (FFHQ) dataset, where age serves as the target label and gender is a correlated bias. (4) The **Dogs** & **Cats dataset** [4] is an (animal, color) dataset, where the former and latter represent target and bias attributes, respectively. Each dataset is predominantly composed of biased samples (up to 99.5%), with a minimal presence of bias-conflicting samples. The distribution of bias samples ranges from 95% to 99.5%.

The dimensions of the images differ across the datasets: Colored MNIST has dimensions of 28×28, Corrupted-CIFAR-10^1 has dimensions of 32x32, and the remaining datasets have

dimensions of 224×224. All image data is normalized across each channel using a mean of (0.4914, 0.4822, 0.4465) and a standard deviation of (0.2023, 0.1994, 0.2010). While Colored MNIST does not undergo additional data augmentation, all other datasets benefit from random crop and horizontal flip transformations

Evaluation Protocol: We follow the standard protocol for evaluation as used in the other SOTA methods, where the train and validation sets are similar while the test set is inverted. For example, in training (99% BA dataset and 1% BC) while in testing (99%BC and 1% BA) Our idea is based on the notion of learning non-spurious and inherent features of objects by imposing both dissimilarity constraints in the final layers and similarity in the initial layers. However, for a comprehensive ablation study, we will also include results in below table obtained using only similarity constraints.

1.1 BFFHQ

The BFFHQ dataset [5] is a gender-biased derivative of the widely-used Flickr-Faces-HQ (FFHQ) dataset, with age as the target label and gender as a correlated bias. It predominantly comprises the 'young' category (individuals aged 10 to 29) which is strongly associated with the 'female' gender, while the 'old' category (individuals aged 40 to 59) is linked with the 'male' gender.

1.2 Dogs & Cats

The Dogs & Cats dataset, initially presented by [9] and subsequently restructured by [10] for bias studies from the original training set, features images categorized into "bright dogs and dark cats" versus "dark dogs and bright cats". Within this dataset, the proportion of bias-aligned to bias-conflicting samples is determined as 8037 to 80, which corresponds to a 1% ratio, and 8037 to 452, equating to a 5% ratio. According to [12], neural networks show a predilection for learning shapes over colors, provided the dataset is sufficiently robust. In this context, the dataset stands out from the Colored MNIST dataset in that the shapes are more pronounced, while the color of the animals is de-emphasized.

1.3 Colored MNIST

A modified version of MNIST dataset [8] where we pertubate them systematically with colours that act as bias attributes. Each of the ten digits is intentionally correlated with a particular color (e.g., red for digit 0). To materialize this bias, color is strategically injected into the foreground of each image. Our experimentation with the Colored MNIST dataset, following the methodology established in [10], involves a systematic consideration of different ratios of bias-conflicting samples. For each ratio of interest, the dataset is divided into two subsets: bias-aligned samples and bias-conflicting samples. The distribution of these subsets varies depending on the specific ratio being examined. Here are the exact numbers for each ratio:

- 0.5% ratio: 54751 bias-aligned samples, 249 bias-conflicting samples
- 1% ratio: 54509 bias-aligned samples, 491 bias-conflicting samples
- 2% ratio: 54014 bias-aligned samples, 986 bias-conflicting samples
- 5% ratio: 52551 bias-aligned samples, 2449 bias-conflicting samples.

This deliberate configuration allows us to comprehensively explore and analyze the impact of bias and its conflicts on the performance of machine learning models in the context of Colored MNIST.

1.4 Corrupted CIFAR-10[1]

The second dataset is the perturbed version of original [7]. We adopted the version used by [3], where the data is perturbed with artificially generated distortions, such as fog, adjustments in brightness, or variations in saturation, have been synthetically introduced to each class. These carefully crafted synthetic biases aim to mimic real-world scenarios closely.

1.5 BAR

The Biased Action Recognition (BAR) dataset comprises six action classes, each biased toward distinct environmental contexts. These classes were chosen by examining the imSitu dataset(a dataset supporting situation recognition), which provides action and place labels for still images from Google Image Search. The selected action classes share commonplace characteristics while having distinct place attributes, resulting in six typical action-place pairs: (Climbing, Rock Wall), (Diving, Underwater), (Fishing, Water Surface), (Racing, A Paved Track), (Throwing, Playing Field), and (Vaulting, Sky).

2 Results on BAR dataset

In our experiments with the BAR dataset presented in Table 1, the anticipated performance levels were not achieved. One plausible explanation for this outcome can be traced back to the limited size of the dataset. Specifically, under the 99% bias ratio setting for class "0", there are only two images that correspond to bias-conflicting samples, in stark contrast to the 296 bias-aligned images. The constraints imposed by the diminutive dataset size necessitated the use of pre-trained models. Consequently, our model faces significant limitations in its learning capacity under such conditions, where bias-conflicting samples are extremely scarce. Given all these limitations, our model still gives better performance than baseline [11] as well as recent studies carried out by [1, 5, 10] on 99% bias ratio. However, when evaluated on a 95% bias ratio, we outperform the baseline, although our model's accuracy falls short of the margin in accommodating new developments.

	BAR Dataset [11]	
Bias Ratio(%)	99.00	95.00
Vanilla	70.55	82.53
DisEnt [9]	70.33	83.13
LFF [11]	70.16	82.95
A^2 [1]	71.15	83.07
ReBias [2]	73.01	83.51
Revisiting LFF [10]	73.36	**84.96**
AmpliBias [5]	73.30	84.67
Ours	**74.52**	83.48

Table 1: Comparative accuracy performance (in %) of of various debiasing algorithms on the BAR Dataset.

3 Effect of hyperparameter λ_c

The hyper-parameters can have a profound impact on model performance and need to be adjusted accordingly to get the intended performance. In our method, we adjust the strength

of similarity or dissimilarity, with respect to where it is applied to the F_d model using λ_c. This adjustment is necessary because of some intrinsic factors. Since we are making the layers of bias F_d and debias F_d model dissimilar, it may not always be the case that they need to be completely similar or completely dissimilar. The bias model can have useful information that should be retained by the F_d model to perform well. In fact, to make enhanced predictions on both bias-aligned images, the debias model should also have qualities of the bias model as the debias model is already best in giving the highest performance on bias-aligned images.

For the Coloured MNIST dataset, we see that the highest performance is achieved when λ_c is equal to 0.1. In the case of coloured MNIST, the bias is easy and is learned at early layers; since it is easy it also impacts the F_d model easily. Hence, to debias we need to have a strong λ_c to control the bias and make it dissimilar from F_b. On the other hand, in BFFHQ Dataset, the dissimilarity strength is very low (between 10^{-5} to 10^{-8}). The reason can be attributed to the presence of similar high-level features in both F_d and F_b. Since, the high-level abstractions are very similar for both the models, making them very dissimilar is bound to distort the performance of the model. The same pattern can be seen in Cats & Dogs Dataset where dissimilarity strength is 10^{-8} for a very severe 99% bias aligned dataset and 10^{-3} for 95% biased dataset.

	CMNIST Dataset			
λ_c	99.50	99.00	98.00	95.00
10^{-1}	**66.98**	**78.03**	**84.22**	**88.64**
10^{-2}	60.22	73.41	81.12	86.37
10^{-3}	62.41	74.14	78.36	84.84
10^{-4}	60.81	74.57	79.43	85.81
10^{-5}	64.36	75.18	80.02	84.48
10^{-7}	63.83	76.29	79.51	84.99
10^{-8}	64.57	77.10	80.04	84.39

Table 2: Performance Impact of Varying Dissimilarity Strength λ_c on the Third Layer of a 3-Layer MLP Model on the CMNIST Dataset. The table presents accuracy (in %) across different bias ratios in the dataset, illustrating the effect of λ_c on robustness against bias.

	BFFHQ Dataset			
λ_c	99.50	99.00	98.00	95.00
10^{-1}	82.20	78.80	88.40	86.40
10^{-2}	81.60	80.00	86.40	89.20
10^{-3}	82.20	78.40	87.00	88.60
10^{-4}	79.80	80.00	88.40	88.80
10^{-5}	81.40	79.50	88.00	**90.20**
10^{-6}	**83.20**	**82.20**	86.80	88.60
10^{-8}	81.40	78.70	**88.40**	87.00

Table 3: Performance Impact of Varying Dissimilarity Strength λ_c on the second MLP layer of pre-trained ResNet18 with three MLP layers on the BFFHQ dataset. The last MLP layer corresponds to the classification layer. The table presents accuracy (in %) across different bias ratios in the dataset, illustrating the effect of λ_c on model robustness against bias.

	Cats & Dogs Dataset	
λ_c	99.00	95.00
10^{-1}	90.40	95.00
10^{-2}	89.90	95.60
10^{-3}	89.80	**96.50**
10^{-4}	89.20	94.10
10^{-5}	90.70	94.10
10^{-6}	92.20	94.80
10^{-8}	**93.00**	95.00

Table 4: Performance with varying λ_c on the second MLP layer of pre-trained ResNet18 with three MLP layers on the Cats & Dogs dataset. The table presents accuracy (in %) across different bias ratios in the dataset.

4 Effect on bias aligned and bias conflicting Samples

Our model -CosFairNet demonstrates superior performance on both unbiased and bias-conflicting samples in comparison to the vanilla model. The latter tends to acquire biases, which are simpler to assimilate than the true underlying signals, and thus, it predominantly relies on these biases for classification. As depicted in Figure 1, the vanilla model exhibits near-perfect accuracy in recognizing bias-aligned samples. However, it significantly underperforms in accurately classifying bias-conflicting and overall unbiased samples, highlighting the effectiveness of our approach in mitigating bias. Furthermore, it is important to note that classifying bias-aligned samples is not as critical as identifying bias-conflicting samples. Therefore, despite the vanilla model performing exceptionally well on bias-aligned samples, its practical value is limited.

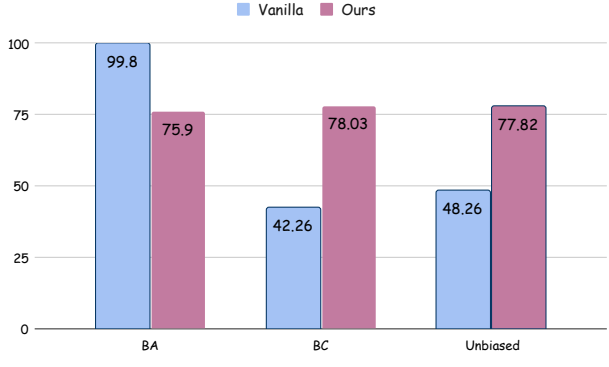

Figure 1: Comparative Analysis of Model Performance on Bias-Aligned, Unbiased, and Bias-Conflicting Samples. In this figure, 'BA' denotes Bias-Aligned, and 'BC' denotes Bias-Conflicting samples. The unbiased dataset has images from both the BA and BC sets.

5 Effect of layer selection

Biases come in different magnitudes, with some being easy to address and others more challenging. Depending on the ease of bias within the dataset, they are learned at either the early layers, mid-layers, or the final layers of the deep learning model. In Table 5, we present an ablation study of applying similarity or dissimilarity at different layers of the model. We

observe a few things - first, applying layer similarity at just one layer works very well and yields optimal results, whereas applying it to multiple layers leads to a drastic degradation in performance. For instance, applying similarity or dissimilarity to the second layer of a three-layer dense network results in very high accuracy, surpassing the present state-of-the-art. However, when applied to both the first and second layers, the performance drops to less than half of the intended accuracy. Additionally, controlling bias at earlier layers seems to be generally more effective compared to controlling bias at later layers of the model.

	CMNIST						
	1	**2**	**3**	**12**	**13**	**23**	**123**
Similarity	74.3	75.13	73.83	30.05	41.24	69.85	26.76
Dissimilarity	65.11	**78.03**	76.31	28.45	30.57	72.54	27.94

Table 5: Performance Comparison of applying Similarity and Dissimilarity on Different Layers of a 3-Layer MLP model on the CMNIST Dataset with 99% Bias Ratio. This table represents the impact of applying similarity and dissimilarity metrics to the first (1), second (2), and third (3) layers, as well as their combinations (12, 13, 23, 123), elucidating how interventions in specific layers affect the overall model performance.

Figure 2 demonstrates our model's performance improvement over time, especially when employing both similarity and dissimilarity constraints, surpassing other models in effectiveness and convergence. This highlights the importance of layer-specific orthogonality in strengthening debiasing efforts.

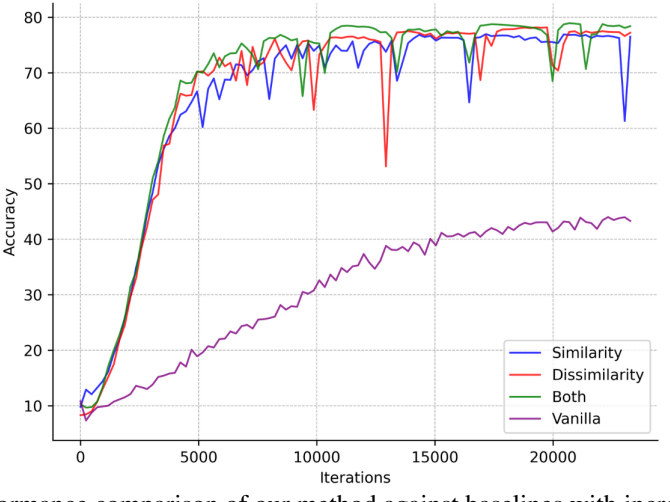

Figure 2: Performance comparison of our method against baselines with increasing training iteration. The results are shown on the CMNIST dataset that has a 99% bias ratio."Both" refers to our final model after the combined application of similarity and dissimilarity constraints. (Best view in colour).

6 Grad-CAM Visualization

In Figure 3, using Grad-CAM visualizations, we present a comparative analysis between our proposed model- CosFairNet and the vanilla model in the context of the BFFHQ dataset [5].

The GradCam visualizations elucidate the regions within the images that the models focus on to infer age, which serves as the target label, amidst the gender-biased data. It is evident from the 'Vanilla' model columns that the attention is unevenly distributed across gender lines. On the other hand, our model demonstrates a more uniform attention distribution, indicating a generalized approach towards learning gender when determining age. This is particularly noteworthy in images where the 'old' category overlaps with 'female' characteristics and the 'young' with 'male', showcasing our model's improved capability to generalize beyond the biased associations present in the dataset.

Figure 3: Grad-CAM Visualization Comparing Vanilla and Our Model's Attention on the BFFHQ Dataset. The left column, 'Vanilla', shows a concentration on gendered features while 'Ours' on the right demonstrates a more equitable distribution of attention, emphasizing our model's resistance to gender bias.

7 Algorithm

The proposed algorithm 1, CosFairNet, aims to train a debiased model by leveraging cosine similarity to realign model parameters. The training process involves a biased model $F_b(x;\theta_b)$ and a debiasing model $F_d(x;\theta_d)$, both initialized with their respective parameters. For each mini-batch of labeled samples, predictions are generated from both models, followed by the computation of cross-entropy loss for the debiasing model and generalized

cross-entropy loss for the biased model. The biased model parameters are updated using gradient descent. The difficulty score is then obtained to weight the loss for updating the debiasing model. Additionally, a dissimilarity loss is introduced, which uses cosine similarity to ensure the initial layers of both models are aligned while the later layers diverge. This alignment aims to reduce bias by penalizing similarity in later layers, thereby promoting diversity in learned features. The algorithm iteratively updates the debiased model parameters, ultimately yielding a model with minimized bias and enhanced generalization.

Algorithm 1: CosFairNet: Parameters Realignment using Cosines

Input: Training data $\mathcal{D} = \{(x_1, y_1), ..., (x_N, y_N)\}$
Parameters θ_b and θ_d, learning rate η, cosine strength λ_c, Initialize biased $F_b(x; \theta_b)$ and debiasing $F_d(x; \theta_d)$ model

Output: Trained debiased model $F(; \hat{\hat{\theta}}_d)$

for *mini batch of labeled samples* $(x_i, y_i) \in \mathcal{D}$ **do**
  Prediction from bias model: $\hat{y}_i^b \leftarrow F_b(x_i, \theta_b)$;
  Prediction from debias model: $\hat{y}_i^d \leftarrow F_d(x_i, \theta_d)$;
  Losses: $\mathcal{L}_d \leftarrow \text{CE}(\hat{y}_i^d; y_i)$
    $\mathcal{L}_b \leftarrow \text{GCE}(\hat{y}_i^b; y_i)$
  Update the biased model:
  $\hat{\theta}_b \leftarrow \theta_b - \eta \frac{\partial \mathcal{L}_b}{\partial \theta_b}$
  Obtain $\mathcal{W}(x_i)$ from Eq. **??**
  Update the debiased model:
  $\hat{\theta}_d \leftarrow \theta_d - \eta \frac{\partial \mathcal{W}(x_i) \cdot \mathcal{L}_d}{\partial \theta_d}$
  Calculate the dis-similarity loss:
  for k^{th} *layer parameters* $\hat{\theta}_{b(k)} \in \hat{\theta}_b$ *and* $\hat{\theta}_{d(k)} \in \hat{\theta}_d$ **do**
    if *k is initial layer* **then**
      $\lambda_c \mathcal{L}_{cosSim}(\hat{\theta}_{d(k)}, \hat{\theta}_{b(k)})$ from Eq. **??**
      $\hat{\hat{\theta}}_d \leftarrow \hat{\theta}_d - \eta \frac{\partial \mathcal{L}_{cosSim}}{\partial \hat{\theta}_d}$;
    else
      $\mathcal{L}_{dis} \leftarrow \lambda_c \left(1 - \mathcal{L}_{cosSim}(\hat{\theta}_{d(k)}, \hat{\theta}_{b(k)})\right)$
      $\hat{\hat{\theta}}_d \leftarrow \hat{\theta}_d - \eta \frac{\partial \mathcal{L}_{dis}}{\partial \hat{\theta}_d}$;
    end
  end
end

The code for the proposed model "CosFairNet" and additional details can be found on the project page: https://visdomlab.github.io/CosFairNet/

References

[1] Jaeju An, Taejune Kim, Donggeun Ko, Sangyup Lee, and Simon S Woo. Aˆ 2: Adaptive augmentation for effectively mitigating dataset bias. In *Proceedings of the Asian Conference on Computer Vision*, pages 4077–4092, 2022.

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

[3] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

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

[5] Eungyeup Kim, Jihyeon Lee, and Jaegul Choo. Biaswap: Removing dataset bias with bias-tailored swapping augmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14992–15001, 2021.

[6] Donggeun Ko, Dongjun Lee, Namjun Park, Kyoungrae Noh, Hyeonjin Park, and Jaekwang Kim. Amplibias: Mitigating dataset bias through bias amplification in few-shot learning for generative models. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, CIKM '23, page 4028–4032, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701245. doi: 10.1145/3583780.3615184. URL https://doi.org/10.1145/3583780.3615184.

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

[8] Yann LeCun, Corinna Cortes, Chris Burges, et al. Mnist handwritten digit database, 2010.

[9] Jungsoo Lee, Eungyeup Kim, Juyoung Lee, Jihyeon Lee, and Jaegul Choo. Learning debiased representation via disentangled feature augmentation. *Advances in Neural Information Processing Systems*, 34:25123–25133, 2021.

[10] Jungsoo Lee, Jeonghoon Park, Daeyoung Kim, Juyoung Lee, Edward Choi, and Jaegul Choo. Revisiting the importance of amplifying bias for debiasing. *AAAI-23*, 5 2022. URL http://arxiv.org/abs/2205.14594.

[11] Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33:20673–20684, 2020.

[12] Samuel Ritter, David GT Barrett, Adam Santoro, and Matt M Botvinick. Cognitive psychology for deep neural networks: A shape bias case study. In *International conference on machine learning*, pages 2940–2949. PMLR, 2017.