Supplementary Material - PASS

Prithviraj Dhar*[1], Joshua Gleason*[2], Aniket Roy[1], Carlos D. Castillo[1], Rama Chellappa[1]
[1]Johns Hopkins University, [2]University of Maryland, College Park
{pdhar1,aroy28,carlosdc,rchella4}@jhu.edu, gleason@umd.edu
* These authors have contributed equally to this work.

In this supplementary material, we provide information about the following: 1. Relation between attribute predictability and bias (Sec. 1), 2. Detailed algorithm (pseudocode) for PASS and MultiPASS (Sec. 2), 3. Hyperparameters used for training PASS and MultiPASS systems (Sec. 3), 4. Hyperparameters for training IVE systems (Sec. 4), 5. Our hair-obscuring pipeline (similar to [1]) (Sec. 5), 6. Detailed results (including verification plots) for de-biasing methods applied on Arcface/Crystalface descriptors (Sec. 6), 7. Ablation study for PASS systems (Sec. 7), 8. Effect of training a discriminative embedding (TPE[14]) on face descriptors and their PASS counterpart (Sec. 8), 9. Advantages of deploying PASS system over end-to-end training (Sec. 9), 10. Discussion about the trade-off between bias reduction and drop in verification performance (Sec. 10).

1. Relation between predictability and bias

In the Section 3 of the main paper, we hypothesize that *reducing the ability to predict protected attributes (gender and skintones) in face descriptors will reduce gender/skintone bias in face verification tasks*. This hypothesis is built on the results of [7], which shows that adversarially removing sensitive information from face representations reduces bias. In the context of gender/skintone bias, we conduct additional experiments to provide the reasoning for this hypothesis. We compare the gender and skintone predictability (i.e. ability to classify an gender/skintone) of face descriptors extracted from Arcface and Crystalface networks and analyze the corresponding bias demonstrated by these networks.

Evaluating gender bias and predictability: Using the IJB-C dataset, we first build a training set with 60k images (30k males and females). Similarly, we construct a test set of 20k images (10k males and females). The images for training and testing are selected randomly, and the face descriptors are extracted using the pre-trained networks (Arcface or Crystalface). There is no overlap between the identities in training and testing set. Subsequently, we train an MLP classifier on face descriptors of the training set to classify gender and evaluate it on the test descriptors. This is done for both Arcface and Crystalface descriptors. The MLP classifier is a two hidden layer MLP with 128 and 64 hidden units respectively with SELU activations, followed by a sigmoid activated output layer. The gender classification accuracy is reported in Table A1. Using the gender-wise verification results in Figure 2(a) in the main paper, we also compute the gender bias at every FPR and present it in Table A1.

FPR		10^{-5}			10^{-4}			10^{-3}		
Network	Acc-g	TPR_m	TPR_f	Bias	TPR_m	TPR_f	Bias	TPR_m	TPR_f	Bias
Arcface	82.06	0.921	0.900	**0.021**	0.962	0.947	**0.015**	0.969	0.956	0.013
Crystalface	86.73	0.836	0.806	0.030	0.913	0.867	0.046	0.952	0.944	0.008

Table A1. Gender bias in IJB-C verification - Arcface vs Crystalface. Acc-g = performance of MLP classifier in predicting Gender.

FPR		10^{-4}			10^{-3}			10^{-2}		
Network	Acc-s	TPR_l	TPR_d	Bias	TPR_l	TPR_d	Bias	TPR_l	TPR_d	Bias
Arcface	87.15	0.951	0.938	**0.013**	0.974	0.968	**0.006**	0.976	0.974	**0.002**
Crystalface	89.30	0.912	0.864	0.048	0.948	0.921	0.027	0.974	0.963	0.011

Table A2. Skintone bias in IJB-C verification - Arcface vs Crystalface. Acc-s = performance of MLP classifier in predicting Skintone.

Evaluating skintone bias and predictability: We follow the same experimental setup for skintone. The only difference is that the training and testing sets are balanced in terms of skintone (dark, medium and light) and the MLP has three output nodes corresponding to light, medium, and dark skintones. The skintone classification accuracy is reported in Table A2. Using the skintone-wise verification results in Figure 2(b) in the main paper, we also compute the skintone bias at every FPR and present it in Table A2.

From the results in Tables A1 and A2, we find that Arcface descriptors have lower gender/skintone predictability than Crystalface descriptors. Moreover, the Arcface descriptors also demonstrate lower gender/skintone bias than their Crystalface counterparts at most FPRs (Tables A1 and A2). From this, we infer that *face descriptors with low gender/skintone predictability appear to demonstrate lower gender/skintone bias in face verification*, thus forming the basis of our initial hypothesis. Therefore, we propose techniques and construct baselines to reduce the predictability of gender and skintone in face descriptors while making them proficient in identity classification.

Algorithm 1 PASS

Required: N_{ep}: Number of training episodes
Required: $\lambda, K, T_{fc}, A^*, T_{deb}, T_{atrain}, T_{plat}, T_{ep}$
Required Learning rates: $\alpha_1, \alpha_2, \alpha_3$
for i in **range**(N_{ep}) **do**
  Begin **Stage 1** (initial training of M and C)
  if i == 0 **then**
    Initialize ϕ_M and ϕ_C with random weights
    for n in **range**(T_{fc}) **do**
      $\phi_M \longleftarrow \phi_M - \alpha_1 \nabla_{\phi_M} L_{class}(\phi_M, \phi_C)$
      $\phi_C \longleftarrow \phi_C - \alpha_1 \nabla_{\phi_C} L_{class}(\phi_M, \phi_C)$
    end for
  end if
  Begin **Stage 2** (initial training of E)
  if i mod T_{ep} == 0 **then**
    Initialize ϕ_E with random weights
    for n in **range**(T_{atrain}) **do**
      $\phi_E \longleftarrow \phi_E - \alpha_2 \nabla_{\phi_E} L_{att}(\phi_M, \phi_E)$
    end for
  end if
  Begin **Stage 3** (update M and C)
  for n in **range**(T_{deb}) **do**
    $\phi_M \longleftarrow \phi_M - \alpha_3 \nabla_{\phi_M} L_{br}(\phi_C, \phi_M, \phi_E)$
    $\phi_C \longleftarrow \phi_C - \alpha_3 \nabla_{\phi_C} L_{br}(\phi_C, \phi_M, \phi_E)$
  end for
  Begin **Stage 4** (update E_k)
  $k = i$ mod K
  for n in **range**(T_{plat}) **do**
    Compute validation attribute prediction accuracy A of E_k
    if $A > A^*$ **then**
      break
    end if
    $\phi_{E_k} \longleftarrow \phi_{E_k} - \alpha_2 \nabla_{\phi_{E_k}} L_{att}^{(E_k)}(\phi_M, \phi_{E_k})$
  end for
end for

Why reduce predictability of protected attributes? Reducing predictability of a protected attribute from a face descriptor to zero implies that no information about that attribute is present in the descriptor. This also implies that no information about the attribute is used to represent identity. Thus, following from the data processing inequality [4], any prediction that is a function of the descriptor is independent of the protected attribute.

2. PASS and MultiPASS algorithm

In section 4.1.1 of the main paper, we explain the components of our proposed adversarial PASS system and discuss the stage-wise training procedure in section 4.1.2 (main paper). Here, we present the detailed algorithm for PASS in Algorithm 1.

Following this, we extend PASS to MultiPASS by reducing the information of two attributes simultaneously: Attribute a, with $N_{att}^{(a)}$ categories and attribute b, with $N_{att}^{(b)}$ categories. The detailed algorithm for training MultiPASS is provided in Algorithm 2. We include two ensembles of discriminators in MultiPASS: one for attribute a, denoted as $E^{(a)}$ and one for attribute b, denoted as $E^{(b)}$. Let $E^{(a)}$ and $E^{(b)}$ consist of K_a and K_b adversary classifiers respectively. The weights for all the classifiers in $E^{(a)}$ are collectively denoted as $\phi_{E^{(a)}}$ and those for $E^{(b)}$ are denoted as $\phi_{E^{(b)}}$. The stage 1 training for model M in MultiPASS is same as that in PASS.

Stage 2: In stage 2, we train both $E^{(a)}$ (for $T_{atrain}^{(a)}$ itera-

Algorithm 2 MultiPASS

Required: N_{ep}: Number of training episodes
Required:$\lambda_a, \lambda_b, K_a, K_b, T_{fc}, A_1^*, A_2^*$
Required:$T_{deb}, T_{atrain}^{(a)}, T_{atrain}^{(b)}, T_{plat}, T_{ep}$
Required Learning rates: $\alpha_1, \alpha_2, \alpha_3$
for i in **range**(N_{ep}) **do**
  Begin **Stage 1** (initial training of M and C)
  if i == 0 **then**
    Initialize ϕ_M and ϕ_C with random weights
    for n in **range**(T_{fc}) **do**
      $\phi_M \longleftarrow \phi_M - \alpha_1 \nabla_{\phi_M} L_{class}(\phi_M, \phi_C)$
      $\phi_C \longleftarrow \phi_C - \alpha_1 \nabla_{\phi_C} L_{class}(\phi_M, \phi_C)$
    end for
  end if
  Begin **Stage 2** (initial training of $E^{(a)}, E^{(b)}$)
  if i mod T_{ep} == 0 **then**
    Initialize $\phi_{E^{(a)}}, \phi_{E^{(b)}}$ with random weights
    for n in **range**($T_{atrain}^{(a)}$) **do**
      $\phi_{E^{(a)}} \longleftarrow \phi_{E^{(a)}} - \alpha_2 \nabla_{\phi_E} L_{att}^{(a)}(\phi_M, \phi_{E^{(a)}})$
    end for
    for n in **range**($T_{atrain}^{(b)}$) **do**
      $\phi_{E^{(b)}} \longleftarrow \phi_{E^{(b)}} - \alpha_2 \nabla_{\phi_E} L_{att}^{(b)}(\phi_M, \phi_{E^{(b)}})$
    end for
  end if
  Begin **Stage 3** (update M and C)
  for n in **range**(T_{deb}) **do**
    $\phi_M \longleftarrow \phi_M - \alpha_3 \nabla_{\phi_M} L_{br}(\phi_C, \phi_M, \phi_{E^{(a)}}, \phi_{E^{(b)}})$
    $\phi_C \longleftarrow \phi_C - \alpha_3 \nabla_{\phi_C} L_{br}(\phi_C, \phi_M, \phi_{E^{(a)}}, \phi_{E^{(b)}})$
  end for
  Begin **Stage 4** (update $E_{k_a}^{(a)}, E_{k_b}^{(b)}$)
  $k_a = i$ mod K_a
  $k_b = i$ mod K_b
  for n in **range**(T_{plat}) **do**
    Compute validation attribute prediction accuracy A_1 of $E_{k_a}^{(a)}$ and A_2 of $E_{k_b}^{(b)}$
    if $A_1 > A_1^*$ and $A_2 > A_2^*$ **then**
      break
    end if
    $\phi_{E_{k_a}^{(a)}} \longleftarrow \phi_{E_{k_a}^{(a)}} - \alpha_2 \nabla_{\phi_{E_{k_a}^{(a)}}} L_{att}^{(E_{k_a}^{(a)})}(\phi_M, \phi_{E_{k_a}^{(a)}})$
    $\phi_{E_{k_b}^{(b)}} \longleftarrow \phi_{E_{k_b}^{(b)}} - \alpha_2 \nabla_{\phi_{E_{k_b}^{(b)}}} L_{att}^{(E_{k_b}^{(b)})}(\phi_M, \phi_{E_{k_b}^{(b)}})$
  end for
end for

tions) and $E^{(b)}$ (for $T^{(b)}_{atrain}$ iterations). An adversarial classifier $E^{(a)}_k$ in $E^{(a)}$ is trained with a standard cross entropy classification loss $L^{E^{(a)}_k}_{att}$

$$L^{E^{(a)}_k}_{att} = -\sum_{i=1}^{N^{(a)}_{att}} y_{a,i} \log\, y^{(k)}_{a,i}. \quad (1)$$

Here $\mathbf{y_a}$ denotes the one hot label with respect to attribute a. $\mathbf{y^{(k)}_a}$ is the softmaxed output from the k^{th} adversary classifier in ensemble $E^{(a)}$. The classification loss $L^{(a)}_{att}$ (in line 17 of Algorithm 2) for the entire ensemble $E^{(a)}_k$ is computed by summing up $L^{E^{(a)}_k}_{att}$ as follows:

$$L^{(a)}_{att} = \sum_{k=1}^{K_a} L^{E^{(a)}_k}_{att} \quad (2)$$

We train the classifiers in ensemble $E^{(b)}$ in a similar way.
Stage 3: Subsequently, we train model M for T_{deb} iterations to generate f_{out} to classify identities (similar to stage 3 in Algorithm 1), while reducing the information of attributes a and b simultaneously. f_{out} from M is provided to both $E^{(a)}$ and $E^{(b)}$ for computing debiasing losses $L^{(a)}_{deb}$ and $L^{(b)}_{deb}$ (See Eq. 14 in main paper). This is used to compute the bias reducing classification loss L_{br} (Eq 15 in the main paper).
Stage 4: After stage 3, we update the adversary classifiers in $E^{(a)}$ and $E^{(b)}$. Using our proposed OAT strategy we choose one classifier $E^{(a)}_{k_a}$ in $E^{(a)}$ and $E^{(b)}_{k_b}$ in $E^{(b)}$ (Lines 29 and 30 in Algorithm 2). We train them for T_{plat} iterations or until $E^{(a)}_{k_a}$ reaches a threshold accuracy of A^*_1 and $E^{(b)}_{k_b}$ reaches a threshold accuracy of A^*_2 on the validation set. We run stages 3 and 4 alternatively, for T_{ep} episodes, after which we re-initialize and re-train all the models in $E^{(a)}$ and $E^{(b)}$ (as done in stage 2).

3. Hyperparameters for PASS and MultiPASS

We provide the hyperparameters used to train PASS-g and PASS-s systems on Arcface and Crystalface descriptors in Table A3.

In our MultiPASS framework, we use attribute a as gender ($N^{(a)}_{att} = 2$, male/female), and attribute b as race ($N^{(b)}_{att} = 4$, Caucasian/Indian/Asian/African). Thus $E^{(a)}$ is an ensemble of gender classifiers and $E^{(a)}$ is an ensemble of race classifiers. Note that, we train MultiPASS on BUPTBalancedFace which consists of race labels, since we currently do not have a large training dataset with skintone labels. The hyperparameters for MultiPASS systems are provided in Table A4. We use a batch size of 400 in all the experiments.

4. Hyperparameters for IVE(g) and IVE(s)

IVE [17] is an attribute suppression algorithm that uses a decision tree ensemble to score each variable in face representations with respect to their importance for a specific recognition task. Variables affecting attribute classification in a significant way are then excluded from the representation. Each step of exclusion removes n_e variables from the representation. The algorithm runs for n_s steps, thus resulting in removal of $n_s \times n_e$ variables from the representation. We train IVE(g) by using face descriptors of MS1M dataset, extracted using a pre-trained netowrk (Arcface or Crystalface). The gender labels are obtained using [12].

We follow the same experimental setup for training IVE(s). The only difference is that the training dataset for training IVE(s) is BUPT-BalancedFace [18]. The official implementation for training IVE is publicly available [16]. In all of our IVE experiments, we use the parameters values mentioned in the code, i.e. $n_s = 20$ and $n_e = 5$, thus resulting in 100 eliminations. Since face descriptors from Arcface or Crystalface are 512-dimensional, the trained IVE(s/g)

Network		Arcface		Crystalface	
Hyperparameter	Stage	PASS-g	PASS-s	PASS-g	PASS-s
λ	3	10	10	1	10
K	2, 3, 4	3	2	4	2
T_{fc}	1	10000	10000	16000	16000
T_{deb}	3	1200	1200	1200	1200
T_{atrain}	2	30000	30000	30000	30000
T_{plat}	4	2000	2000	2000	2000
A^*	4	0.95	0.95	0.90	0.95
α_1	1	10^{-2}	10^{-2}	10^{-2}	10^{-2}
α_2	2,4	10^{-3}	10^{-3}	10^{-3}	10^{-3}
α_3	3	10^{-4}	10^{-4}	10^{-4}	10^{-4}
T_{ep}	3,4	40	40	40	40

Table A3. Hyperparameters for training PASS-g and PASS-s on Arcface and Crystalface descriptors

Hyperparameter	Stage	Arcface	Crystalface
λ_a	3	10	1
λ_b	3	10	10
K_a	2, 3, 4	3	4
K_b	2, 3, 4	2	2
T_{fc}	1	10000	16000
T_{deb}	3	1200	1200
$T^{(a)}_{atrain}$	2	30000	30000
$T^{(b)}_{atrain}$	2	30000	30000
T_{plat}	4	2000	2000
A^*_1	4	0.95	0.90
A^*_2	4	0.95	0.95
α_1	1	10^{-2}	10^{-2}
α_2	2,4	10^{-3}	10^{-3}
α_3	3	10^{-4}	10^{-4}
T_{ep}	3,4	40	40

Table A4. Hyperparameters for training MultiPASS on Arcface and Crystalface descriptors

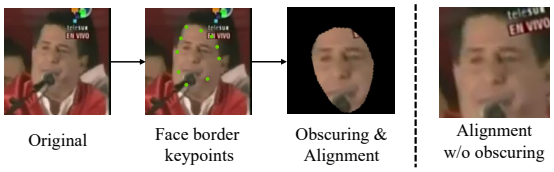


Figure A1. Our method for obscuring hair (Similar to [1]). On the right, we show an aligned image without obscuring hair.

framework transforms the input descriptors for test images into $512 - 100 = 412$ dimensional descriptors. These descriptors are then used to perform face verification.

5. Hair obscuring - Similar to [1]

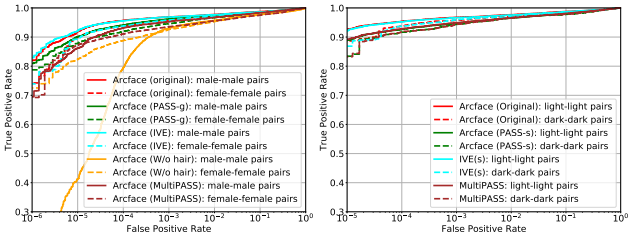
In [1], it is shown that after obscuring hair in facial images, the resulting face descriptors extracted using Arcface demonstrate lower gender bias. However, such experiments are only performed on datasets with clean frontal faces in MORPH [13] and Notre-Dame [10] datasets. The authors used a segmentation network [20] to obscure the hair. But, in complex datasets, e.g., IJB-C containing varied and cluttered poses, segmenting out hair region is non-trivial and hard to perform. Instead, we compute the face border keypoints using [12] and obscure all the regions outside the polygon formed by these keypoints. Our hair obscuring pipeline is presented in Fig A1. Note that, [1] proposes hair-obscuring as a possible approach to specifically mitigate gender-bias, and not skintone bias. So, we do not evaluate the effect of hair-obscuring while analyzing skintone bias.

6. Detailed results

6.1. PASS with Arcface

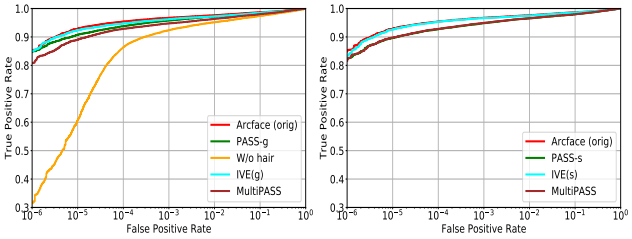
For PASS/MultiPASS systems trained on Arcface descriptors, we provide the gender-wise and skintone-wise results in Table 2 and 3 respectively in the main paper. We also present the gender and skintone bias in Figure 6 in the main paper, and show that the PASS/MultiPASS systems outperform the IVE and hair-obscuring baselines at most FPRs. Here, we provide the gender-wise and skintone-wise verification plots for all the methods used to de-bias Arcface descriptors in Figure A2. Additionally, we also provide the overall verification plots in Figure A3.

Although the main aim of using PASS-g is to reduce gender predictability in face descriptors, we find (in Fig. A2a) that the performance of female-female verification improves between FPR 10^{-5} and 10^{-6}. In fact, we find several examples of template pairs which are verified between these FPRs, for both Arcface descriptors and their PASS-g counterparts. In such pairs, we find the average cosine similarity of images in templates that belong to the same female identity increases after the face descriptors are transformed using PASS-g. We



(a) (b)

Figure A2. (a.) Gender-wise and (b.) Skintone-wise verification plots for Arcface descriptors and their de-biased counterparts on IJB-C



(a) (b)

Figure A3. Overall IJB-C verification plots of Arcface along with (a.) Gender-debiasing algorithms, (b.) Skintone-debiasing algorithms.

Figure A4. Examples of templates in IJB-C verification for which the average cosine similarity improved after PASS transformation.

show two examples of such templates in Fig A4.

6.2. PASS with Crystalface

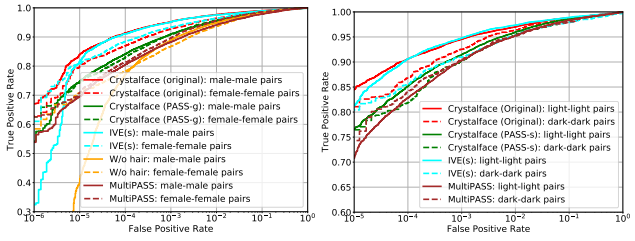
It can be inferred from Tables A1 and A2 that descriptors from Crystalface demonstrate higher gender/skintone bias than those from Arcface. Therefore, we believe that debiasing Crystalface descriptors is a better testing ground for de-biasing algorithms like PASS/MultiPASS. Moreover, this helps us assess the generalizability of proposed PASS/MultiPASS systems. We provide the BPC values and overall TPRs of all the approaches for de-biasing Crystalface descriptors in Table 4 (for gender) and Table 5 (for skintone) in the main paper, and show that PASS/MultiPASS systems achieve higher BPC values than the baselines. Here, we provide the gender-wise and skintone-wise verification TPRs (along with the corresponding bias values) in Tables A5 and A6 respectively. Moreover, we provide the gender-wise and skintone-wise verification plots for all the methods in Figure A5. Also, we provide the overall verification plots for all the methods in Figure A6. It should be noted in Tables A5

FPR	10^{-5}					10^{-4}					10^{-3}				
Network	TPR_m	TPR_f	TPR	Bias (↓)	BPC_g (↑)	TPR_m	TPR_f	TPR	Bias (↓)	BPC_g (↑)	TPR_m	TPR_f	TPR	Bias (↓)	BPC_g (↑)
Crystalface[11]	0.836	0.806	0.833	0.030	0.000	0.913	0.867	0.910	0.046	0.000	0.952	0.924	0.951	0.028	0.000
W/o hair[1]	0.424	0.713	0.589	0.289	-8.926	0.774	0.779	0.809	0.005	0.780	0.881	0.875	0.899	0.006	0.731
IVE(g)[17]	0.818	0.813	0.833	0.005	0.833	0.912	0.884	0.910	0.028	0.391	0.952	0.926	0.951	0.026	0.071
PASS-g	0.751	0.749	0.761	**0.002**	**0.847**	0.831	0.828	0.839	**0.003**	**0.857**	0.909	0.909	0.910	**0.00**	**0.956**
MultiPASS	0.699	0.713	0.708	0.014	0.383	0.811	0.808	0.809	**0.003**	0.823	0.879	0.883	0.881	0.004	0.784

Table A5. *Gender* bias analysis of *Crystalface* descriptors, and their transformed counterparts on IJB-C. TPR: overall True Positive rate, TPR_m: male-male TPR, TPR_f: female-female TPR. **Bold**=Best, Underlined=Second best

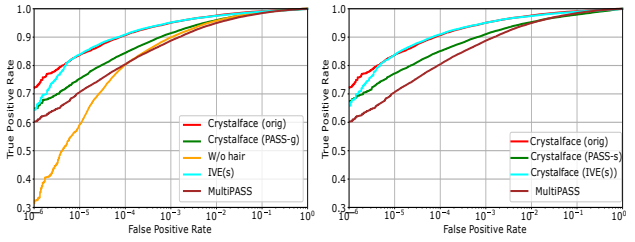
FPR	10^{-4}					10^{-3}					10^{-2}				
Network	TPR_l	TPR_d	TPR	Bias (↓)	BPC_{st} (↑)	TPR_l	TPR_d	TPR	Bias (↓)	BPC_{st} (↑)	TPR_l	TPR_d	TPR	Bias (↓)	BPC_{st} (↑)
Crystalface[11]	0.912	0.864	0.910	0.048	0.000	0.948	0.921	0.951	0.027	0.000	0.974	0.963	0.974	0.011	0.000
IVE(s)[17]	0.912	0.862	0.910	0.050	-0.041	0.949	0.911	0.951	0.038	-0.407	0.975	0.953	0.974	0.022	-1.000
PASS-s	0.850	0.818	0.844	0.032	0.261	0.913	0.906	0.914	0.007	0.702	0.962	0.953	0.919	0.009	0.125
MultiPASS	0.826	0.838	0.809	**0.012**	**0.639**	0.907	0.907	0.881	**0.000**	**0.927**	0.953	0.953	0.968	**0.000**	**0.994**

Table A6. *Skintone* bias analysis of *Crystalface* descriptors, and their transformed counterparts on IJB-C. TPR: overall True Positive rate, TPR_l: light-light TPR, TPR_d: dark-dark TPR. **Bold**=Best, Underlined=Second best



(a) (b)

Figure A5. (a.) Gender-wise and (b.) Skintone-wise verification plots for Arcface descriptors and their de-biased counterparts on IJB-C



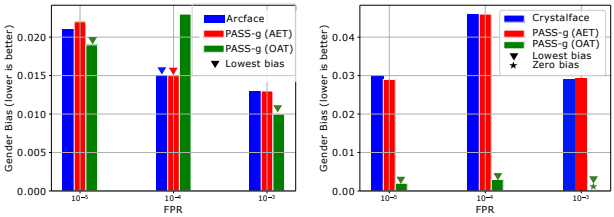
(a) (b)

Figure A6. Overall IJB-C verification plots of Crystalface along with (a.) Gender-debiasing algorithms, (b.) Skintone-debiasing algorithms.

and A6 that although IVE achieves higher overall TPRs, it hardly reduces bias, thus obtaining lower BPC values than PASS/MultiPASS systems.

6.3. OAT v/s AET

In Figure A7, we visualize the results presented in Table 7 in the main paper.



(a) (b)

Figure A7. Comparison of bias for AET vs OAT in gender reduction on (a) Arcface, (b) Crystalface.

6.4. Results with multiple skintones

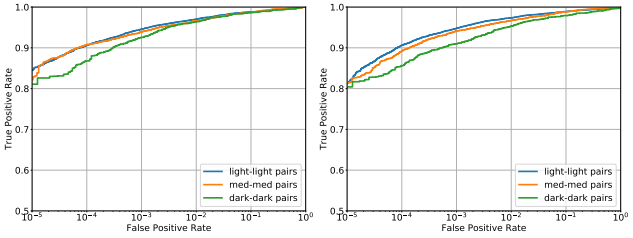
In Equations 1 and 2 in the main paper, we define bias as the absolute difference between the verification TPRs of two groups at a given FPR. However, it possible that a sensitive attribute consists of more than two categories. For instance, the skintone attribute consists of three categories: Light, medium, dark. In the main paper, we chose to define bias as the difference between the verification TPRs of light-light and dark-dark pairs at a given FPR. However, as shown in [18], we can also define bias as the standard deviation (STD) among the verification TPRs of light-light pairs, medium-medium pairs and dark-dark pairs. In Table A7, we report these STD values for our PASS-s and MultiPASS systems (and the corresponding baselines) trained on Crystalface descriptors, along with the average of the TPRs obtained for the three skintone categories. *We find that our proposed PASS-s/MultiPASS systems obtain considerably lower STD than existing baselines, thus mitigating skintone bias.* We also provide the skintone-wise verification plots for all three skintones (light, medium and dark) on IJB-C dataset in Figure A8

FPR	10^{-4}					10^{-3}					10^{-2}				
Method	TPR_l	TPR_{med}	TPR_d	Avg	STD (↓)	TPR_l	TPR_{med}	TPR_d	Avg	STD (↓)	TPR_l	TPR_{med}	TPR_d	Avg	STD (↓)
Crystalface	0.912	0.912	0.864	0.896	0.023	0.948	0.939	0.921	0.936	0.011	0.974	0.964	0.963	0.967	0.005
IVE(s)	0.912	0.899	0.862	0.891	0.021	0.949	0.946	0.911	0.935	0.017	0.975	0.968	0.953	0.965	0.009
PASS-s (ours)	0.850	0.861	0.818	0.843	<u>0.018</u>	0.913	0.909	0.906	0.909	<u>0.003</u>	0.962	0.957	0.953	0.957	<u>0.004</u>
MultiPASS (ours)	0.826	0.838	0.838	0.834	**0.006**	0.907	0.908	0.907	0.907	**0.0005**	0.953	0.952	0.953	0.953	**0.0005**

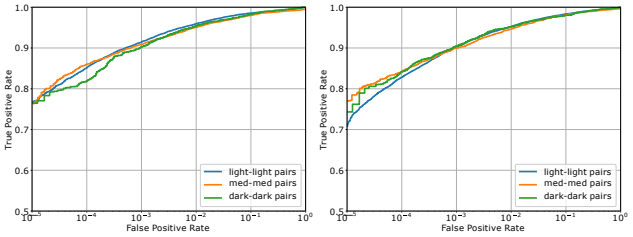
Table A7. Average and Standard deviation (STD) among the verification TPRs of light-light pairs, medium-medium pairs and dark-dark pairs. TPR: overall True Positive rate, TPR_l: light-light TPR, TPR_{med}: medium-medium TPR, TPR_d: dark-dark TPR. **Bold**=Best, <u>Underlined</u>=Second best

FPR	10^{-5}					10^{-4}					10^{-3}				
Method	TPR_m	TPR_f	TPR	Bias (↓)	BPC_g (↑)	TPR_m	TPR_f	TPR	Bias (↓)	BPC_g (↑)	TPR_m	TPR_f	TPR	Bias (↓)	BPC_g (↑)
Crystalface + TPE	0.883	0.838	0.875	0.045	0.000	0.925	0.891	0.924	0.034	0.000	0.962	0.939	0.959	0.023	0.000
PASS-g + TPE	0.797	0.764	0.800	**0.033**	**0.181**	0.875	0.843	0.875	**0.032**	**0.006**	0.929	0.915	0.930	**0.014**	**0.361**

Table A8. IJB-C 1:1 verification results after applying TPE on face descriptors from Crystalface and its PASS-g counterpart. TPR: overall True Positive rate, TPR_m: male-male TPR, TPR_f: female-female TPR.

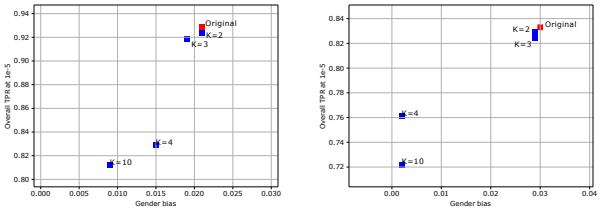


(a) Crystalface (b) IVE(s) on Crystalface

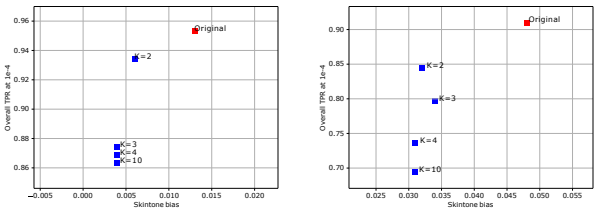


(c) PASS-s on Crystalface (d) MultiPASS on Crystalface

Figure A8. Skintone-wise verification plots for all three skintones on the IJB-C dataset for Crystalface descriptors and their skintone-debiased counterparts



(a) PASS-g on Arcface ($\lambda = 10$) (b) PASS-g on Crystalface ($\lambda = 1$)



(c) PASS-s on Arcface ($\lambda = 10$) (d) PASS-s on Crystalface ($\lambda = 10$)

Figure A9. Effect of varying K (number of adversary classifiers in the ensemble E) in PASS systems

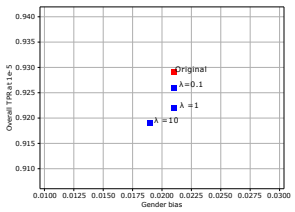
7. Ablation experiments: Effect of K, λ in PASS

In Eq. 11 of the main paper, we combined a classification loss L_{class} and an adversarial de-biasing loss L_{deb} to compute a bias reducing classification loss L_{br} as follows:

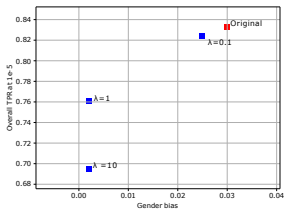
$$L_{br} = L_{class} + \lambda L_{deb} \tag{3}$$

L_{deb} is computed using an ensemble of K attribute classifiers that act as adversaries to model M. λ is the weight applied on this de-biasing loss. Here, we evaluate two hyperparameters used to train the PASS framework : (a) the number of attribute classifiers K in the ensemble E used to compute L_{deb} (Eq. 10 in main paper). (b) the weight λ for L_{deb} defined in Eq. 3 here. We analyze how changing these hyperparameters in PASS-g and PASS-s systems vary the resultant gender bias reduction and verification performance at a fixed FPR in the IJB-C dataset. We perform these experiments on PASS-g and PASS-s trained on both Arcface and Crystalface descriptors. For evaluating the PASS-g systems, we report the gender bias and verification TPR at FPR=10^{-5}. For evaluating PASS-s systems, we report the skintone bias and verification TPR at FPR=10^{-4}. (See Fig. A9 and A10)

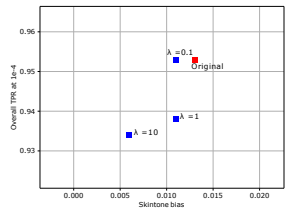
Varying K (number of adversary classifier in the ensemble) : We experiment with $K = 2, 3, 4$ and 10, while fixing all the other parameters. The ablation results for



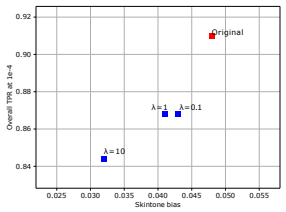
(a) PASS-g on Arcface ($K = 3$)



(b) PASS-g on Crystalface ($K = 4$)

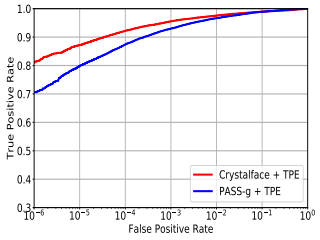


(c) PASS-s on Arcface ($K = 2$)

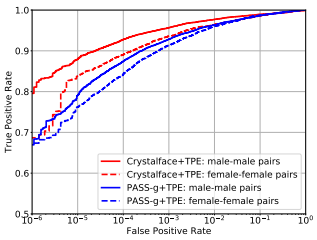


(d) PASS-s on Crystalface ($K = 2$)

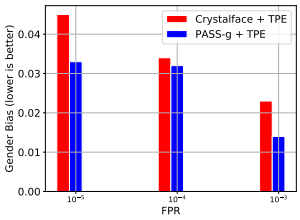
Figure A10. Effect of varying λ (weight for L_{deb}) in PASS systems



(a)



(b)



(c)

Figure A11. (a.) Overall IJB-C verification plots, (b.) Gender-wise IJB-C verification plots, (c.) Associated gender bias for Crystalface descriptors and its PASS-g counterpart after applying TPE

PASS-g systems are presented in Figures A9a (for Arcface) and A9b (for Crystalface). The results for PASS-s systems trained on Arcface descriptors are presented in Figure A9c and those for Crystalface descriptors are presented in Figure A9d. We find that for both PASS-s and PASS-g systems, increasing K generally lowers the corresponding bias but also reduces the verification performance.

Varying λ (weight for L_{deb}): We experiment with $\lambda = 0.1, 1, 10$ for training the PASS-s framework on Arcface and Crystalface descriptors. All the other hyperparameters remain fixed. The results are presented in Fig. A10. For both PASS-g and PASS-s systems, we find that as we keep on increasing the value of λ, the associated bias generally decreases and the verification TPR keeps decreasing.

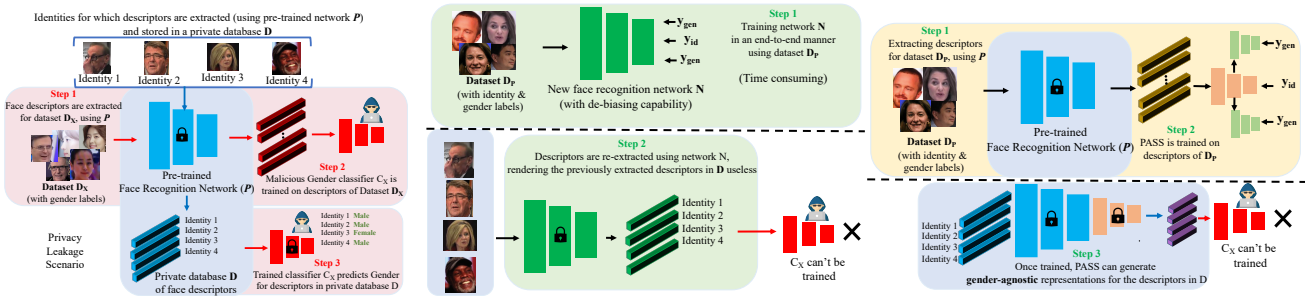
8. Bonus experiment: Effect of TPE

In [11], the face descriptors from Crystalface are not directly used for verification. Instead, the descriptors undergo triplet probabilistic embedding (TPE) [14] for generating a template representation of a given identity. TPE is an embedding learned to generate more discriminative, low-dimensional representations of given input descriptors, that have been shown to achieve better verification results. We apply TPE on the descriptors obtained using Crystalface and find that *TPE improves the overall verification performance, but it also increases gender bias at all FPRs* ('Crystalface + TPE' in Table A8). We analyze if applying TPE on PASS-g descriptors has the same effect. We learn a TPE matrix using Crystalface descriptors transformed with PASS-g. We apply this TPE matrix to transform the PASS-g descriptors extracted for the test (IJB-C) dataset, the results for which are presented in Table A8 ('PASS-g + TPE'). From Table A8 and Figure A11, we can infer that the *gender bias in the verification results obtained after applying TPE on PASS-g transformed descriptors is lower than when TPE is applied on original face descriptors of Crystalface.*

To learn a triplet probabilistic embedding W_{cf}, we use the descriptors from Crystalface (extracted for UMD-Faces [2] dataset). This embedding $W_{cf} \in \mathbb{R}^{512\times128}$ is then used to transform the 512 dimensional IJB-C [9] descriptors (extracted using Crystalface) to obtain 128-dimensional face descriptors, which are used for 1:1 face verification. The results of this experiment are provided in 'Crystalface + TPE' in Table A8. We perform the same experiment with the PASS-g transformed descriptors of Crystalface, where a new TPE matrix $W'_{cf} \in \mathbb{R}^{256\times128}$ is learned and used to transform the IJB-C descriptors before performing 1:1 verification.

For training both, W_{cf} and W'_{cf}, we use a fixed learning rate of 2.5×10^{-3} and a batch size of 32. The training for computing such a matrix using the descriptors from Crystalface (or its PASS-g counterpart) generally converges after 10k iterations. For a given set of descriptors, we compute its TPE matrix ten times and finally compute the average of the resulting matrices. We use this matrix to transform the test descriptors. More details about TPE are provided in [14].

Note that, unlike Crystalface [11], Arcface [5] does not mention applying TPE on the face descriptors and therefore we do not apply TPE on PASS-based systems that are trained



(a) (b) (c)

Figure A12. (a.) Example of a scenario where an agent C_X can cause privacy breach in a private database D that contains a pre-trained face recognition network P and face descriptors of four identities extracted using P. (b.) Training an end-to-end de-biasing system does not allow us to re-use the pre-computed descriptors in D. (c) PASS can be train on top of descriptors from P and can re-use the pre-computed descriptors in D to generate their gender-agnostic representations.

Method	Training	Backbone	#Params w/o final classif^n layer
Debface-ID[7]	End-to-end	ResNet-52	10.99 million
Demo-ID[7]	End-to-end	ResNet-52	10.99 million
GAC[8]	End-to-end	ResNet-52	10.99 million
PASS-g w/ AF	Descriptor-based	MLP	254,336
PASS-s w/ AF	Descriptor-based	MLP	213,504
MultiPASS w/ AF	Descriptor-based	MLP	336,768
PASS-g w/ CF	Descriptor-based	MLP	295,424
PASS-s w/ CF	Descriptor-based	MLP	213,504
MultiPASS w/ CF	Descriptor-based	MLP	377,856

Table A9. Number of trainable parameters in end-to-end and PASS-based methods. AF=Arcface, CF=Crystalface

on Arcface.

9. Advantages of PASS over end-to-end systems

In section 5.4.3 of the main paper, we explained how PASS/MultiPASS systems outperform end-to-end bias mitigation methods like [7] and [8] in terms of overall face verification performance. Apart from this, the PASS/MultiPASS system is easier to deploy than end-to-end pipelines.

Training time: Most end-to-end bias-mitigation techniques ([7] and [8]) use a ResNet architecture, for this reason training such frameworks likely takes a long time. In contrast, our descriptor-based PASS/MultiPASS systems (which are composed of MLPs) have fewer trainable parameters. In Table A9, we compare the number of trainable parameters (excluding the final identity classification layer) of PASS-based systems and other end-to-end debiasing approaches. Since PASS/MultiPASS systems have fewer trainable parameters, the training is relatively fast.

Note that we recognize that convolution layers and linear layers differ in number of floating-point operations per weight, however, we use number of weights here as a rough proxy for computation time.

Re-using pre-computed descriptors: We go back to the example scenario described in Fig 1 of the main paper (and here in Fig A12a). Suppose a malicious agent X has gained access to a private database D (blue) which consists of a pre-trained network P and face descriptors of four identities. The agent can use P to extract descriptors (red) for a gender-labeled dataset D_X (Step 1). Using these descriptors, the agent can train a gender classifier C_X (Step 2). Using the trained C_X, the agent can predict the gender of the descriptors in D (Step 3) and thus cause privacy breach.

Let's say we apply an end-to-end bias mitigation technique to prevent such privacy breach (Fig A12b). We first need to train a network N on a dataset with identity and gender labels. This step is time consuming. Also, once N is trained, we need to re-extract the face descriptors for the four identities using N. Thus, the pre-computed descriptors in D cannot be re-used.

Instead, suppose that we deploy PASS-g for this task (Fig A12c). We can use the pre-trained network P to first extract face descriptors for a dataset with identity and gender labels. Using these descriptors, we can train a PASS-g system. Once trained, PASS-g can be quickly applied to the pre-computed descriptors to generate their gender agnostic representations. This re-use of existing descriptors is not possible using an end-to-end de-biasing system. Thus, compared to end-to-end de-biasing methods, PASS allows easier deployment.

10. A discussion about bias reduction and drop in verification performance

Although PASS/MultiPASS systems are trained to reduce sensitive information from face descriptors while maintaining their identity classification capability, it is clear from Figures A3 and A6 that reducing information of sensitive attributes in face descriptors leads to a slight drop in verification performance. This is not unexpected because attributes like gender and race/skintone are entangled with identity [6], and are integral to it . Hence, reducing the information of such attributes is expected to slightly reduce the face descriptors' ability to classify identities. In fact, several works that reduce information of sensitive attributes demonstrate a

drop in overall performance of the system. For instance, [3] proposes a method to suppress gender in face representations while performing the task of face recognition. Although this method successfully enhances gender privacy in the representations, it also leads to a slight drop in face recognition performance. Similarly, [19] proposes a method to perform activity recognition while reducing sensitive identity information. However, this leads to a slight drop in the target task of activity recognition. Also, [15] proposes a GAN-based framework to generate a dataset that is fair (neutral) in terms of gender and skintone, while performing the target task of predicting attractiveness. While this method reduces the gender/skintone bias in attractiveness prediction, this also leads to a slight drop in the attractiveness prediction accuracy.

References

[1] V Albiero and KW Bowyer. Is face recognition sexist? no, gendered hairstyles and biology are. *BMVC*, 2020. 1, 4, 5

[2] A Bansal, A Nanduri, C D Castillo, R Ranjan, and R Chellappa. Umdfaces: An annotated face dataset for training deep networks. In *2017 IEEE International Joint Conference on Biometrics (IJCB)*, pages 464–473. IEEE, 2017. 7

[3] B Bortolato, M Ivanovska, P Rot, J Križaj, Philipp Terhörst, Naser Damer, Peter Peer, and Vitomir Štruc. Learning privacy-enhancing face representations through feature disentanglement. In *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)(FG)*, pages 45–52. IEEE Computer Society, 2020. 9

[4] T M Cover and J A Thomas. *Elements of information theory*. John Wiley & Sons, 2012. 2

[5] J Deng, J Guo, X Niannan, and S Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, 2019. 7

[6] P Dhar, A Bansal, CD Castillo, J Gleason, PJ Phillips, and R Chellappa. How are attributes expressed in face dcnns? In *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, pages 85–92. IEEE, 2020. 8

[7] S Gong, X Liu, and AK Jain. Jointly de-biasing face recognition and demographic attribute estimation. In *European Conference on Computer Vision*, pages 330–347. Springer, 2020. 1, 8

[8] S Gong, X Liu, and AK Jain. Mitigating face recognition bias via group adaptive classifier. In *In Proceeding of IEEE Computer Vision and Pattern Recognition*, Nashville, TN, June 2021. 8

[9] B Maze, J Adams, J A Duncan, N Kalka, T Miller, C Otto, A K Jain, W T Niggel, J Anderson, J Cheney, et al. IARPA janus benchmark-c: Face dataset and protocol. In *2018 International Conference on Biometrics (ICB)*, pages 158–165. IEEE, 2018. 7

[10] PJ Phillips, PJ Flynn, T Scruggs, KW Bowyer, J Chang, K Hoffman, J Marques, J Min, and W Worek. Overview of the face recognition grand challenge. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 1, pages 947–954. IEEE, 2005. 4

[11] R Ranjan, A Bansal, J Zheng, H Xu, J Gleason, B Lu, A Nanduri, J-C Chen, C D Castillo, and R Chellappa. A fast and accurate system for face detection, identification, and verification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 1(2):82–96, 2019. 5, 7

[12] R Ranjan, S Sankaranarayanan, C D Castillo, and R Chellappa. An all-in-one convolutional neural network for face analysis. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 17–24. IEEE, 2017. 3, 4

[13] K Ricanek and T Tesafaye. Morph: A longitudinal image database of normal adult age-progression. In *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, pages 341–345. IEEE, 2006. 4

[14] S Sankaranarayanan, A Alavi, C D Castillo, and R Chellappa. Triplet probabilistic embedding for face verification and clustering. In *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, 2016. 1, 7

[15] P Sattigeri, SC Hoffman, V Chenthamarakshan, and KR Varshney. Fairness gan: Generating datasets with fairness properties using a generative adversarial network. *IBM Journal of Research and Development*, 63(4/5):3:1–3:9, 2019. 9

[16] P Terhörst, N Damer, F Kirchbuchner, and A Kuijper. Ive code. https://github.com/pterhoer/PrivacyPreservingFaceRecognition/tree/master/supervised/incremental_variable_elimination, 2019. 3

[17] P Terhörst, N Damer, F Kirchbuchner, and A Kuijper. Suppressing gender and age in face templates using incremental variable elimination. In *2019 International Conference on Biometrics (ICB)*, pages 1–8. IEEE, 2019. 3, 5

[18] M Wang and W Deng. Mitigating bias in face recognition using skewness-aware reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9322–9331, 2020. 3, 5

[19] Z Wu, Z Wang, Z Wang, and H Jin. Towards privacy-preserving visual recognition via adversarial training: A pilot study. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 606–624, 2018. 9

[20] C Yu, J Wang, C Peng, C Gao, G Yu, and N Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 325–341, 2018. 4