Supplementary Material for SHASAM: Submodular Hard Sample Mining for Fair Facial Attribute Recognition

Anay Majee
The University of Texas at Dallas
anay.majee@utdallas.edu

Rishabh Iyer
The University of Texas at Dallas
rishabh.iyer@utdallas.edu

Contents

1. Notation

Following the problem definition in the main paper we introduce the notations used in Tab. 1 throughout the paper.

2. Additional Related Work and Preliminaries

2.1. Contrastive Learning

In the realm of supervised learning, conventional models utilizing Cross-Entropy (CE) loss [1] often grapple with challenges posed by class imbalance and noisy labels. To address these issues, metric learning techniques [4, 29, 35, 36] aim to learn distance-based [30] or similarity-based [4, 35] metrics, fostering orthogonality within the feature space [29] and bolstering class-specific feature discrimination. Contrastive learning, rooted in noise contrastive estimation [8], has become a cornerstone in self-supervised learning [2, 3, 9], where label information is unavailable during training. In supervised contexts, SupCon [15] emphasizes forming feature clusters rather than merely aligning features to predefined centroids. For instance, Triplet loss [30] differentiates one positive and one negative pair, whereas N-pairs [32] loss incorporates multiple negative pairs, and SupCon extends this by leveraging multiple positive and negative pairs. Lifted-Structure loss [33] sharpens focus by contrasting positives against the hardest negatives, and SupCon exhibits similarities with Soft-Nearest Neighbors loss [6], which maximizes inter-class entanglements. Despite the significant achievements of these methods, they predominantly rely on pairwise similarity metrics, which may not inherently facilitate the formation of disjoint clusters.

2.2. Submodularity (Cont. from Section 3.2)

As discussed in Sec. 3.2 of the main paper, submodular functions have been recognized to model notions of cooperation [11], diversity [19], representation [18] and coverage [13]. Following the combinatorial formulation in Sec.3.1 of the main paper we define the ground set $\mathcal{V} = \{T_1, T_2, \cdots T_N\} = \{S_1, S_2, \cdots S_K\}$ and explore four different categories of submodular information functions in our work, namely -

(1) *Submodular Total Information* (S_f) which measures the total information contained in each set [7], expressed as $S_f(T_1, T_2, \ldots, T_N)$ as in Eq. (1). Maximizing S_f over

Table 1. Collection of notations used in the paper.

Symbol	Description
$\mathcal{V}$	The Ground set, here refers to the mini-batch at each iteration.
T_i	The target attribute set $T_i, \forall i \in \lvert\mathcal{V}\rvert$.
S_j	The sensitive attribute set $S_j, \forall j \in \lvert\mathcal{V}\rvert$.
A_{ij}	Anchor set with examples from target attribute T_i and sensitive attribute S_j.
A_p^i	Hard-Positives with examples from target attribute T_i and sensitive attribute $\overline{S_j}$.
A_n^i	Hard-Negatives with examples from target attribute $\mathcal{V} \setminus T_i$ and sensitive attribute S_j.
$F(x, \theta)$	Neural Network used as feature extractor.
$Clf(.,.)$	Multi-Layer Perceptron as classifier. In our case a two layer network.
θ	Parameters of the feature extractor.
$S_{A,B}(\theta)$	Cross-Similarity between sets $A, B \in \mathcal{V}$.
$S_A(\theta)$	Self-Similarity between samples in set $A \in \mathcal{T}$.
$f(A)$	Submodular Information function over a set A.
$I_f(A;Q)$	Submodular Mutual Information function between sets A and Q.
$H_f(A\lvert Q)$	Submodular Conditional Gain function between sets A and Q.
$I_f(A;Q\lvert P)$	Submodular Conditional Mutual Information function between a target set A, query set Q and private set P.
$L_{\text{SHASAM}}(\theta)$	Loss value computed over all target and sensitive attribute pairs.
$N_f(A_{ij})$	Normalization constant approximated to $3\lvert A_{ij}\rvert$.
EO	Equalized Odds.

a set T_i models diversity [19] while minimizing S_f models cooperation [11].

$$S_f(T_1, T_2, \ldots, T_N) = \sum_{i=1}^{N} f(T_i) \tag{1}$$

(2) *Submodular Mutual Information* (I_f) which models the shared information between two sets [18] which serves as a measure of *similarity/cooperation* between them, expressed through Eq. (2).

$$I_f(T_i; T_j) = f(T_i) + f(T_j) - f(T_i \cup T_j), \forall i, j \in |\mathcal{V}| \tag{2}$$

(3) *Submodular Conditional Gain* (H_f) which models the gain in information when a set T_j is added to T_i. H_f models the notion of *dissimilarity* between sets and can be expressed in Eq. (3).

$$\begin{aligned} H_f(T_i|T_j) &= f(T_i \cup T_j) - f(T_j) \\ &= f(T_i) - I_f(T_i; T_j)\ , \forall i, j \in |\mathcal{V}| \end{aligned} \tag{3}$$

(4) *Submodular Conditional Mutual Information* (I_f) which jointly models the mutual similarity between two sets T_i and T_j and their collective dissimilarity to a conditioning set C as:

$$\begin{aligned} I_f(T_i; T_j|C) &= f(T_i \cup C) + f(T_j \cup C) \\ &\quad - f(T_i \cup T_j \cup C) - f(C) \\ I_f(T_i; T_j|C) &= I_f(T_i \cup C; T_j) - I_f(T_j; C) \\ &= H_f(T_i|C) + H_f(T_j|C) \\ &\quad - H_f(T_i \cup T_j|C)\ , \forall i, j \in |\mathcal{V}| \end{aligned} \tag{4}$$

Note, that the above formulations can also be reformulated by considering the sensitive attribute S_i instead of T_i , $\mathcal{V} = \cup_{i=1}^{N} T_i = \cup_{j=1}^{K} S_j$. Given a submodular function f (can alternatively be I_f or H_f) tasks like selection [10, 16] and summarization [13, 14] have been modeled as a discrete optimization problem to identify a summarized set of examples $A \subseteq \mathcal{V}$ via submodular maximization under a cardinality constraint ($|A| \leq k$), i.e. $max_{A \subseteq \mathcal{V}, |A| \leq k} f(A)$. This can be fairly approximated with a $(1 - e^{-1})$ constant factor guarantee [23] using greedy optimization techniques [22] as shown in Sec. 2.2. Extending the definition of submodular functions to continuous optimization space Majee et al. [20] have proposed a set of novel family of learning objectives which minimize total information and total correlation among sets in D_{train} using continuous optimization techniques like SGD. These objectives have been shown to be significantly more robust to large imbalance demonstrated in real-world tasks like longtail recognition [20] and few-shot learning [21].

3. Additional Explanation to Fig. 1

In this section we elaborate the steps of operation of SHASAM which is condensed and depicted in Fig. 1 of the main paper. At first, given a ground set (a large labeled pool) $\mathcal{V}$ we would like to mine an anchor set A_{12}. Anchor sets are supposed to contain examples sharing the same target and sensitive attributes T_1 and S_2 as shown in Fig. 1. In our example T_1 refers to males and $\overline{T_1}$ refers to non-males. Similarly S_2 refers to males wearing sunglasses and $\overline{S_2}$ refers to people (irrespective of gender) that do not wear sunglasses. Thus, A_{12} contains examples of males wearing eyeglasses. Once we have the anchor set,

Algorithm 1 Greedy Submodular Maximization as in Nemhauser et al. [23].

Require: Submodular function $f : 2^{\mathcal{V}} \rightarrow \mathbb{R}$, cardinality constraint k

Ensure: Set $A \subseteq \mathcal{V}$ maximizing $f(A)$ under cardinality constraint k

Initialize an empty set $A \leftarrow \emptyset$
for $j = 1$ to k **do**
  $e \leftarrow \underset{v \in \mathcal{V} \setminus A}{\texttt{argmax}}(f(A \cup \{v\}) - f(A))$
  $A \leftarrow A \cup \{e\}$
end for
return A

we now use the formulation described in Sec. 3.3 we mine A_p^1 which resembles the hard-positives containing examples of males which do not wear eyeglasses. Similarly we now mine hard-negatives A_n^1 which contain examples of females (or $\overline{T_1}$) who wear eyeglasses as shown in Fig. 1(a). This is the operation of SHASAM-MINE.

Once we have mined A_{12}, A_p^1 and A_n^1 we now would like to learn embedding separation between A_{12} and An^1 while bringing A_p^1 closer to A_{12}. As shown in Fig. 1(b) we apply SHASAM-LEARN which is a loss function derived from Submodular Conditional Mutual Information (SCMI) as shown in eq. 3 of the main paper. Following the discussion in Sec. 3.5 SCMI based learning objectives jointly model anchor-hard-negative separation while modeling anchor-hard-positive cooperation when minimized used SGD. With sufficient training SHASAM-LEARN results in an embedding space which learns the decision boundary between males and non-males without biasing on the sensitive attribute - wearing sunglasses as shown in Fig. 1(b).

4. Additional Implementation Details

The implementation details are largely elucidated in Sec. 4 of the main paper but we add more details below and at https://anaymajee.me/assets/project_pages/shasam.html.

4.1. Settings for Selection in SHASAM-MINE

As discussed in Algorithm 1 of the main paper we employ three different submodular functions to sample the anchors A_{ij}, hard-positives A_p^i and hard-negatives A_n^i. We run the selection at every iteration of the training process with 16 anchors, 16 hard-positives and 16 hard-negatives selected in each iteration. Each image further gets augmented into two views which results in a batch size of 96 as discussed in Sec. 4 of the main paper. Following our problem formulation, at each iteration we randomly select a target attribute T_i and a sensitive attribute S_j where $i, j \in |\mathcal{V}|$. All anchors in A_{ij} share the same target and sensitive attribute, thus mined from $T_i \cap S_j$, while the hard-positives share the same target attribute but different sensitive attribute labels from $T_i \cap \overline{S_j}$. Finally, negatives are mined from $\overline{T_i} \cap S_j$ and share orthogonal target attributes while retaining the same sensitive attribute. This formulation follows Park et al. [26] and results in selection of samples at the cluster boundary.

Since the ground set resembles the complete dataset at each iteration, selection of A_{ij}, A_p^i and A_n^i becomes computationally inefficient (large computational cost). To mitigate this situation, we randomly select a subset of the dataset at each epoch and use it for model training. This strategy draws inspiration from Okanovic et al. [24] and significantly improves training speeds without compromising on performance.

4.2. Training and Evaluation in SHASAM-LEARN

As discussed in Sec. 3 of the main paper, SHASAM-MINE and SHASAM-LEARN are sequentially invoked at each iteration for training the feature extractor F on dataset D^{train}. Following the experimental setup in Sec. 4 of the main paper, each image in both CelebA and UTKFace datasets are resized to 128 $\times$ 128 pixels with two complementary augmentations among - RandomCrop, Grayscaling, ColorJitters etc. similar to Khosla et al. [15]. For the implementations of GRL [27], LNL [17], FD-VAE [25], MFD [12], FSCL [26] we follow the implementation in Park et al. [26] and report the average Top-1 Accuracy (Acc.) and Equalized Odds (EO) over three random seeds. Particularly in case of FSCL, we adopt their two-stage training strategy to train a ResNet-18 based feature extractor on the train split for 100 epochs with a 0.1 initial learning rate and a cosine annealing scheduler. On the contrary, our SHASAM approach is trained with a higher initial learning rate of 0.4 with a cosine annealing scheduler for $\sim$ 20 epochs during stage 1. For stage 2 both FSCL and SHASAM are trained for 10 epochs, keeping the feature extractor frozen.

For FairViT [34] we adopt a ViT [5] based feature extractor with fairness driven modifications as suggested in the original paper. We perform a single stage training till convergence. Unlike using the first 80 individuals from the CelebA dataset as in the original implementation, we use all individuals available in CelebA for reproduction of the results. For fair comparisons, we replace the feature extractor in SHASAM (ResNet-18) with a vanilla ViT architecture and conduct two sets of experiments, one following FSCL (denoted as FSCL w/ViT in Table 2) and the other adopting the SHASAM (denoted as SHASAM w/ FLCMI, ViT in Table 2) with Facility-Location Conditional Mutual Information (FLCMI) based learning objective.

4.3. Experiments on Synthetic Dataset

To characterize the selection strategy discussed in Sec. 3.4 (SHASAM-MINE) we conduct experiments on synthetic

datasets by varying (1) *imbalance* and (2) *Feature Similarity* (cluster overlap). To this end we introduce a 2-dimensional 2-cluster setting as shown in Fig. 4 of the main paper. For varying imbalance we ablate on the imbalance ratio α from 1 in the balanced setting and 5 in the imbalanced setting. Fig. 4 (main paper) depicts the imbalanced setting with $\alpha = 5$. The abundant cluster A consists of 500 examples with high intra-group variance while the rare class has 100 examples with low intra-group variance. The selection budget was set to 10 examples for each anchor, hard-positive and hard-negative sets. To contrast against *Random* sampling (commonly used strategy in [15, 26, 31]) we keep the seed constant for underlying libraries. Between the imbalanced settings Fig. 4 (main paper) and Fig. 4 (supplementary), we highlight the difference in selected samples by altering the target attribute between the majority and minority class. For the balanced setting (1), α is set to 1 with each cluster containing 200 examples. This is depicted in Fig. 3. The variance of the clusters indicated through the spread of points in the feature space is also kept constant between cluster A and B. For varying the (2) feature similarity we reduce the separation between two clusters so that some overlap exists between them by reducing the 2D distance between the cluster centroids. The imbalance is kept constant at $\alpha = 5$ and the majority class is considered as the target class,similar to setting in Fig. 4 of the main paper. This setting is depicted through Fig. 5 and discussed in detail in Sec. 8.6.

5. Gradients through $L_{\text{SHASAM}}(\theta)$

In this section we provide proof for our proposed formulation in Sec. 3.2, eq. (2) which states that the gradient through $L_{\text{SHASAM}}(\theta) \approx \nabla L_{\text{SHASAM}}(A_{ij}, A_p^i, A_n^i, \theta)$ which is the gradient over the combinatorial loss alone.

Proof. At first we can consider the definition of $L_{\text{SHASAM}}(\theta)$ as a product of three functions $X(\theta)$, $Y(\theta)$ and $Z(\theta)$ as shown below.

$$
\begin{aligned}
L_{\text{SHASAM}}(\theta) = \sum_{\forall i,j \in |\mathcal{V}|} & \underbrace{\texttt{softmax}(H_f(.|A_{ij}), A_p^i, T_i \cap \overline{S_j}))}_{X(\theta)} \\
& \times \underbrace{\texttt{softmax}(I_f(.; A_{ij}), A_n^i, \mathcal{V} \setminus T_i \cap S_j))}_{Y(\theta)} \\
& \times \underbrace{L_{\text{SHASAM}}(A_{ij}, A_p^i, A_n^i; \theta)}_{Z(\theta)} \\
L_{\text{SHASAM}}(\theta) = \sum_{\forall i,j \in |\mathcal{V}|} & X(\theta) \times Y(\theta) \times Z(\theta)
\end{aligned}
\tag{5}
$$

Given this simplified form, we calculate the gradients through $L_{\text{SHASAM}}(\theta)$ following the chain rule as -

$$
\begin{aligned}
\frac{\partial L_{\text{SHASAM}}(\theta)}{\partial \theta} = & \frac{\partial X(\theta)}{\partial \theta} \times Y(\theta) \times Z(\theta) \\
& + X(\theta) \times \frac{\partial Y(\theta)}{\partial \theta} \times Z(\theta) \\
& + X(\theta) \times Y(\theta) \times \frac{\partial Z(\theta)}{\partial \theta}
\end{aligned}
\tag{6}
$$

We know that, $X(\theta)$ and $Y(\theta)$ are hard-positive and hard-negative miners and are approximated as the $\texttt{softmax}$ over the selected subset A from Q given a selection function $F(.)$. Lets call this σ -

$$
\sigma_i(\theta) = \texttt{softmax}(F(.;\theta), A, Q) = \frac{\exp(F_i(A;\theta))}{\sum_j \exp(F_j(A;\theta))} \tag{7}
$$

Irrespective of the choice of the selection function $F(.)$ the gradient over σ can be written as -

$$
\frac{\partial \sigma_i(\theta)}{\partial F_i(A)} = \begin{cases} \sigma_i(\theta)(1 - \sigma_i(\theta)) & \text{if } i = k, \\ -\sigma_i(\theta)\sigma_k(\theta) & \text{if } i \neq k. \end{cases} \tag{8}
$$

In the first case, when $i == k$ the softmax function evaluates to 1 and occurs when $\sigma_i(\theta)$ is approximately the $\texttt{argmax}$. Similarly, for the second case, $i \neq k$ occurs when $\sigma_i(\theta)$ does not approximate the argmax. In this case $\sigma_i(\theta)$ evaluates to 0. Thus, in both cases the gradients through $X(\theta)$ and $Y(\theta)$ approximate to 0. This reduces the gradient calculation for $L_{\text{SHASAM}}(\theta)$ as -

$$
\frac{\partial L_{\text{SHASAM}}(\theta)}{\partial \theta} = X(\theta) \times Y(\theta) \times \frac{\partial Z(\theta)}{\partial \theta} \tag{9}
$$

Since $Z(\theta)$ represents the loss function $L_{\text{SHASAM}}(A_{ij}, A_p^i, A_n^i, \theta)$ and thus:

$$
\nabla_\theta L_{\text{SHASAM}}(\theta) \approx \nabla_\theta L_{\text{SHASAM}}(A_{ij}, A_p^i, A_n^i, \theta) \tag{10}
$$

Where C is a constant, $C = X(\theta).Y(\theta)$. This proves our consideration in the formulation in Sec. 3.2 and allows SHASAM to utilize a mixture of discrete and continuous optimization problems to model learning of fair representations as a submodular hard sample mining. □

6. Derivation of Instances of SHASAM

We define the loss function in SHASAM as the conditional mutual information between the mined anchors A_{ij}, hard-positives A_p^i and hard-negatives A_n^i as shown in eq. (3) of the main paper. We summarize it here for readability as Eq. (11).

$$
L_{\text{SHASAM}}(\theta) = \sum_{\forall i,j \in |\mathcal{V}|} \frac{1}{N_f(A_{ij})} I_f\left(A_{ij}; A_n^i | A_p^i; \theta\right) \tag{11}
$$

In this section we derive two important instances of L_{SHASAM} based on the choice of the submodular function $f(A, \theta)$ over set A as summarized in Table 1 of the main paper.

6.1. SHASAM-FLCMI

Given a dataset $\mathcal{V}$ and the Facility-Location (FL) submodular function $f(A) = \sum_{i \in \mathcal{V}} \max_{j \in A} S_{ij}$ over a set A, we derive a combinatorial loss L_{SHASAM} on the mined anchors A_{ij}, hard-positives A_p^i and hard-negatives A_n^i based on the Submodular Conditional Mutual Information function $I_f(A_{ij}; A_n^i | A_p^i)$ as shown in Eq. (12).

$$L_{\text{SHASAM}}(\theta) = \sum_{i,j \in |\mathcal{V}|} \frac{1}{3|A_{ij}|} \max\left(\min\left(\max_{a \in A_{ij}} S_{ia}, \max_{n \in A_n^i} S_{in}\right) - \max_{p \in A_p^i} S_{ip}, 0\right) \tag{12}$$

Proof. From the definition of CMI based on conditional gain as shown in Eq. (4),

$$\begin{aligned} I_f(A_{ij}; A_n^i | A_p^i) =& H_f(A_{ij} | A_p^i) + H_f(A_{ij} | A_n^i) \\ & - H_f(A_{ij} \cup A_n^i | A_p^i) \\ =& f(A_{ij}) - I_f(A_{ij}; A_p^i) \\ & + f(A_{ij}) - I_f(A_{ij}; A_n^i) \\ & - f(A_{ij} \cup A_n^i) + I_f(A_{ij} \cup A_n^i; A_p^i) \end{aligned} \tag{13}$$

Now, we separate each term containing I_f and expand them based on the definition of SMI in Eq. (2) -

$$\begin{aligned} I_f(A_{ij}; A_p^i) =& f(A_{ij}) + f(A_p^i) - f(A_{ij} \cup A_p^i) \\ =& \sum_{i,j \in \mathcal{V}} \max_{a \in A_{ij}} S_{ia} + \sum_{i,j \in \mathcal{V}} \max_{p \in A_p^i} S_{ip} \\ & - \sum_{i,j \in \mathcal{V}} \max_{k \in A_{ij} \cup A_p^i} S_{ik} \\ =& \sum_{i,j \in \mathcal{V}} \max_{a \in A_{ij}} S_{ia} + \max_{p \in A_p^i} S_{ip} \\ & - \max\left(\max_{k \in A_{ij}} S_{ik}, \max_{k \in A_p^i} S_{ik}\right) \\ =& \sum_{i,j \in \mathcal{V}} \min\left(\max_{k \in A_{ij}} S_{ik}, \max_{k \in A_p^i} S_{ik}\right) \end{aligned} \tag{14}$$

Similarly we can calculate $I_f(A_{ij}; A_n^i)$ and $I_f(A_{ij} \cup A_n^i; A_p^i)$ as -

$$I_f(A_{ij}; A_n^i) = \sum_{i,j \in \mathcal{V}} \min\left(\max_{k \in A_{ij}} S_{ik}, \max_{k \in A_n^i} S_{ik}\right) \tag{15}$$

$$I_f(A_{ij} \cup A_n^i; A_p^i) = \sum_{i,j \in \mathcal{V}} \min\left(\max_{k \in A_{ij} \cup A_n^i} S_{ik}, \max_{k \in A_p^i} S_{ik}\right) \tag{16}$$

Substituting these expressions in Eq. (13) and simplifying we get -

$$\begin{aligned} I_f(A_{ij}; A_n^i | A_p^i) =& H_f(A_{ij} | A_p^i) + H_f(A_{ij} | A_n^i) \\ & - H_f(A_{ij} \cup A_n^i | A_p^i) \\ =& \sum_{i,j \in \mathcal{V}} \max\left(0, \underbrace{\max_{k \in A_{ij}} S_{ik}}_{\text{p}} - \underbrace{\max_{k \in A_p^i} S_{ik}}_{\text{r}}\right) \\ & + \max\left(0, \underbrace{\max_{k \in A_n^i} S_{ik}}_{\text{q}} - \underbrace{\max_{k \in A_p^i} S_{ik}}_{\text{r}}\right) \\ & - \max\left(0, \max_{k \in A_{ij} \cup A_n^i} S_{ik} - \max_{k \in A_p^i} S_{ik}\right) \end{aligned} \tag{17}$$

This follows the expression $\max(p-r, 0) + \max(q-r, 0) - \max(\max(p, q) - r, 0)$. Which evaluates to -

$$\begin{cases} \max(q - r, 0) & \text{if } p > q, \\ \max(p - r, 0) & \text{if } p < q \end{cases} \tag{18}$$

Thus we can simplify the expression of $I_f(A_{ij}; A_n^i | A_p^i)$ as -

$$I_f(A_{ij}; A_n^i | A_p^i) = \sum_{i,j \in \mathcal{V}} \max\left(\min\left(\max_{k \in A_{ij}} S_{ik}, \max_{k \in A_n^i} S_{ik}\right) - \max_{k \in A_p^i} S_{ik}, 0\right) \tag{19}$$

Substituting this in the expression of $L_{\text{SHASAM}}(\theta)$ we get -

$$L_{\text{SHASAM}}(\theta) = \sum_{i,j \in |\mathcal{V}|} \frac{1}{3|A_{ij}|} \max\left(\min\left(\max_{k \in A_{ij}} S_{ik}, \max_{k \in A_n^i} S_{ik}\right) - \max_{k \in A_p^i} S_{ik}, 0\right) \text{ ... Hence proved.} \tag{20}$$

□

6.2. SHASAM-LogDetCMI

Given a dataset $\mathcal{V}$ and the Log-Determinant (LogDet) submodular function $f(A) = \text{logdet}(S_A)$ over a set A, we derive a combinatorial loss L_{SHASAM} on the mined anchors

A_{ij}, hard positives A_p^i and hard negatives A_n^i based on the Submodular Conditional Mutual Information function $I_f(A_{ij}; A_n^i | A_p^i)$ as shown in Eq. (21).

$$L_{\text{SHASAM}}(\theta) = \sum_{i,j \in |\mathcal{V}|} \frac{1}{3|A_{ij}|} \log \frac{\det\left(I - S_{A_n^i}^{-1} S_{A_n^i, A_p^i} S_{A_p^i}^{-1} S_{A_n^i, A_p^i}^T\right)}{\det\left(I - S_{A_{ij} \cup A_n^i}^{-1} S_{A_{ij} \cup A_n^i, A_p^i} S_{A_p^i}^{-1} S_{A_{ij} \cup A_n^i, A_p^i}^T\right)} \tag{21}$$

Proof. From the definition of CMI based on Mutual Information as described in Eq. (4), we get -

$$\begin{aligned} I_f(A_{ij}; A_n^i | A_p^i) =& I_f(A_{ij} \cup A_p^i; A_n^i) - I_f(A_n^i; A_p^i) \\ =& f(A_{ij} \cup A_p^i) + f(A_n^i) - f(A_{ij} \cup A_p^i \cup A_n^i) \\ & - f(A_n^i) + f(A_p^i) - f(A_n^i \cup A_p^i) \end{aligned} \tag{22}$$

Given the definition of LogDet over a set A, $f(A) = \text{logdet}(S_A)$, we substitute this in the above equation to get the following -

$$\begin{aligned} I_f(A_{ij}; A_n^i | A_p^i) =& \text{logdet}(S_{A_{ij} \cup A_p^i}) + \text{logdet}(S_{A_n^i}) \\ & - \text{logdet}(S_{(A_{ij} \cup A_p^i) \cup A_n^i}) \\ & - \text{logdet}(S_{A_n^i}) + \text{logdet}(S_{A_p^i}) \\ & - \text{logdet}(S_{A_n^i \cup A_p^i}) \\ =& \log\left(\frac{\det(S_{A_{ij} \cup A_p^i}).\det(S_{A_n^i})}{\det(S_{A_{ij} \cup A_p^i \cup A_n^i})}\right) \\ & - \log\left(\frac{\det(S_{A_n^i}).\det(S_{A_p^i})}{\det(S_{A_n^i \cup A_p^i})}\right) \end{aligned} \tag{23}$$

From Schur's complement we know that $det(S_{A \cup B}) = det(S_A) det(S_{A \cup B} \setminus S_A)$ and $S_{A \cup B} \setminus S_A = S_B - S_{A,B}^T S_A^{-1} S_{A,B}$, where $S_{A,B}$ refers to the cross-similarities between sets A and B while S_A and S_B represent the corresponding self-similarities. We use this to simplify the expression of I_f above as follows -

$$\begin{aligned} I_f(A_{ij}; A_n^i | A_p^i) =& \log\left(\frac{\det(A_n^i)}{\det(S_{A_{ij} \cup A_p^i \cup A_n^i} \setminus S_{A_{ij} \cup A_p^i})}\right) \\ & - \log\left(\frac{\det(S_{A_p^i})}{\det(S_{A_n^i \cup A_p^i} \setminus S_{A_n^i})}\right) \\ =& \log\det\left(\frac{S_{A_n^i \cup A_p^i} \setminus S_{A_n^i}}{S_{A_p^i}}\right) \\ & - \log\det\left(\frac{S_{A_{ij} \cup A_p^i \cup A_n^i} \setminus S_{A_{ij} \cup A_p^i}}{S_{A_n^i}}\right) \\ I_f(A_{ij}; A_n^i | A_p^i) =& \text{logdet}\left(\frac{S_{A_p^i} - S_{A_n^i, A_p^i}^T S_{A_n^i}^{-1} S_{A_n^i, A_p^i}}{S_{A_p^i}}\right) \\ & - \text{logdet}\left(\frac{S_{A_n^i} - S_{A_{ij} \cup A_p^i, A_n^i}^T S_{A_{ij} \cup A_p^i}^{-1} S_{A_{ij} \cup A_p^i, A_n^i}}{S_{A_n^i}}\right) \\ =& \text{logdet}\left(I - S_{A_p^i}^{-1} S_{A_n^i, A_p^i}^T S_{A_n^i}^{-1} S_{A_n^i, A_p^i}\right) \\ & - \text{logdet}\left(I - S_{A_n^i}^{-1} S_{A_{ij} \cup A_p^i, A_n^i}^T S_{A_{ij} \cup A_p^i}^{-1} S_{A_{ij} \cup A_p^i, A_n^i}\right) \end{aligned} \tag{24}$$

Following simple logarithmic principles we can further simplify this expression as -

$$I_f(A_{ij}; A_n^i | A_p^i) = \log \frac{\det\left(I - S_{A_n^i}^{-1} S_{A_n^i, A_p^i} S_{A_p^i}^{-1} S_{A_n^i, A_p^i}^T\right)}{\det\left(I - S_{A_{ij} \cup A_p^i}^{-1} S_{A_{ij} \cup A_p^i, A_n^i} S_{A_n^i}^{-1} S_{A_{ij} \cup A_p^i, A_n^i}^T\right)} \tag{25}$$

Substituting this in the equation of $L_{\text{SHASAM}}(\theta)$ in Eq. (11) we get the loss function for SHASAM-LogDetCMI as shown below.

$$L_{\text{SHASAM}}(\theta) = \sum_{i,j \in |\mathcal{V}|} \frac{1}{3|A_{ij}|} \log \frac{\det\left(I - S_{A_n^i}^{-1} S_{A_n^i, A_p^i} S_{A_p^i}^{-1} S_{A_n^i, A_p^i}^T\right)}{\det\left(I - S_{A_{ij} \cup A_p^i}^{-1} S_{A_{ij} \cup A_p^i, A_n^i} S_{A_n^i}^{-1} S_{A_{ij} \cup A_p^i, A_n^i}^T\right)} \tag{26}$$

□

7. Additional Fairness Metrics (Contd. from Sec. 4)

To provide a comprehensive evaluation of our model, we assess both its predictive performance and its adherence to fairness principles. In addition the reported metrics in Sec. 4 of the main paper we also evaluate our model on additional metrics that have been used in SoTA approaches.

For performance, we utilize **Balanced Accuracy** (BA), a metric particularly effective for datasets with imbalanced class distributions. Rather than a simple accuracy score, BA calculates the average of the true positive and true negative rates across each subgroup, offering a more nuanced view of a model's effectiveness and ensuring that high performance

Table 2. **Ablation on selection budget in SHASAM** measured in terms of Top-1 Accuracy (Acc.) and equalized odds (EO) by varying the budget k on two settings in CelebA dataset. The learning objective was kept constant at SHASAM-LEARN w/ FLCMI and the selection strategy in SHASAM-MINE is set to FLCMI.

Selection Strategy	Budget k	$T = a$ / $S = m$		$T = a$ / $S = y$	
		EO (↓)	Acc. (↑)	EO (↓)	Acc. (↑)
FSCL [26]	-	6.5	79.1	12.4	79.1
SHASAM	8	7.0	79.9	10.2	77.2
SHASAM	16	5.6	81.3	9.9	79.58
SHASAM	24	5.5	81.08	10.1	79.44

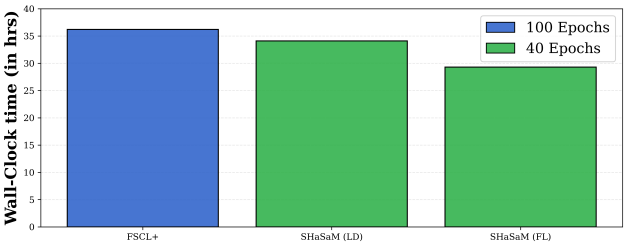

Figure 1. **Ablation on Computational Cost** measured in terms of wall clock time averaged over three settings in CelebA dataset. The submodular function in the learning objective and selection strategy in SHASAM was kept constant.

on a majority group does not obscure poor performance on a minority group.

For fairness, we employ two widely accepted metrics. The first, **Demographic Parity** (DP), measures whether the rate of positive predictions is consistent across different sensitive groups. A model satisfies DP if its decisions are statistically independent of an individual's group membership, with a value approaching zero indicating greater fairness.

The second metric, **Equalized Opportunity** (EOpp), enforces a more stringent fairness condition by requiring that the model's true positive rate is the same for all sensitive groups. This ensures the model is equally effective at correctly identifying positive instances, regardless of group identity. For both DP and EOpp, scores closer to zero signify a more equitable model.

8. Additional Results

8.1. Additional Results from Metrics in Sec. 7

Although Equalized Odds is the most commonly adopted fairness metric in literature there exists metrics such as Demographic Parity (DP), Equalized Opportunity (EOpp) and Balanced Accuracy (BA) that are studied in the context of fair facial attribute recognition. In addition to Equalized Odds (EO) in Tab. 2 (main paper) we present results from the aforementioned metrics by closely following the exact benchmark in FairViT [34] in Tab. 3. Similar to FairViT we also adopt three distinct settings by varying T and S - T = attractiveness (a) / S = gender (male, m), T = expression

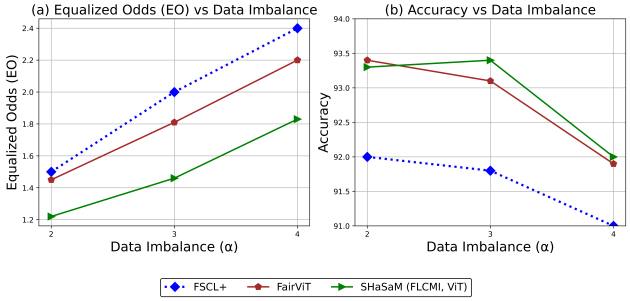

Figure 2. **Results of SHASAM on UTKFace dataset on ViT based benchmark** (FairViT) measuring (a) Equalized Odds and (b) Top-1 Acc. under varying inter-group imbalance (α). The target and sensitive attributes are set to *gender* and *ethnicity* respectively following setup in Park et al. [26].

(smiling, s) / S = gender (male, m), T = attractiveness (a) / S = hair-color (brown, br). Similar to our observations in Tab. 2 (main paper) we see that SHASAM (with FLCMI as the instance for SHASAM-MINE and SHASAM-LEARN) achieves competitive accuracy (indicated as Acc.) to FairViT without the requirement to train the Vision Transformer (ViT) architecture from scratch. Finally, the self-balancing property as elucidated in [20] ensures that majority classes do not bias the decisions made by SHASAM. This is reflected in the boost in performance measured through BA across all downstream tasks.

8.2. Comparison against FairViT on UTKFace Dataset

In continuation to the results presented in Sec. 4.1 on UTKFace dataset we include experiments contrasting the performance of SHASAM against FairViT [34]. For fair comparisons we replace the resnet based backbone in SHASAM with a ViT based backbone and depicts the results in Fig. 2 under similar imbalanced settings discussed in Sec. 4.1 and report the top-1 accuracy and EO through Fig. 2(b) and Fig. 2(a). Similar to Sec. 4.1 we show that our SHASAM w/ FLCMI approach shows improvements in both EO and accuracy under varying degrees of imbalance.

8.3. Ablation on the selection Budget

As discussed in Sec. 4 of the main paper, SHASAM learns from a set of mined anchors, hard-positives and hard-negatives. Since the selection of exemplars in each of these sets is performed through submodular optimization under the knapsack constraint [23], a fixed budget k is established for each set. For simplicity we keep this budget constant across three sets discussed above such that in each iteration there are exactly k anchors, k hard-positives and k hard-negatives resulting in a total batch size of 3k (excluding augmentations). From the results tabulated in Tab. 2 a pattern emerges wherein a higher budget (more examples) benefit both fairness and accuracy metrics. However, its

Table 3. A comparison of different methods across three distinct tasks. The tasks involve predicting an attribute (T) while maintaining fairness with respect to a sensitive attribute (S). We report Accuracy (Acc.), Balanced Accuracy (BA), Equalized Opportunity (EOpp.), and Demographic Parity (DP). Higher ACC and BA are better, while lower EO and DP are better. The best result in each column is highlighted in bold.

method	$T = a$ / $S = m$				$T = s$ / $S = m$				$T = a$ / $S = br$			
	Acc.$_{\%}$	BA$_{\%}$	EOpp.$_{e-2}$	DP$_{e-1}$	Acc.$_{\%}$	BA$_{\%}$	EOpp.$_{e-2}$	DP$_{e-1}$	Acc.$_{\%}$	BA$_{\%}$	EOpp.$_{e-2}$	DP$_{e-1}$
Vanilla	74.01	72.36	14.43	3.245	88.42	88.85	4.91	1.489	76.48	74.55	3.61	1.896
TADeT-MMD [28]	79.89	73.85	7.10	3.693	92.51	93.03	2.48	1.290	77.97	75.64	2.27	1.491
TADeT [28]	78.73	74.52	3.11	3.116	90.05	90.68	4.86	1.443	78.49	77.42	3.78	1.057
FSCL [26]	79.09	74.76	1.78	3.004	89.37	90.08	1.76	1.344	78.85	78.06	2.65	0.989
FSCL+ [26]	77.26	73.42	**0.79**	2.604	88.83	89.02	1.20	1.263	78.02	77.37	1.79	0.834
FairViT [34]	83.80	79.96	1.15	2.837	**94.27**	94.12	1.52	1.205	**82.52**	**81.56**	2.10	**0.701**
SHASAM(w/ FLCMI, ViT) (ours)	**84.01**	**80.16**	0.76	**2.582**	92.87	**94.57**	**1.17**	**1.193**	80.70	79.42	**2.06**	0.873

interesting to note that the performance saturates beyond $\texttt{k} = 16$. Due to this (alongside compute limitations) we adopt a selection budget of 16 in all our experiments. Note, we were unable to execute experiments with higher selection budgets in Tab. 2 due to compute limitations.

8.4. Ablation: Choice of Combinatorial Function in SHASAM

As discussed in Sec. 3.4 and 3.5 the choice of submodular function f induces various combinatorial properties encoded in SHASAM-MINE and SHASAM-LEARN. In Tab. 3 of the main paper we vary the submodular function between LD (indicating LogDetCMI) and FL (indicating FLCMI) functions and report the performance both interms of EO and accuracy (indicated as Acc. in Tab. 3). At first, we show that FL based selection and learning functions demonstrate improved performance and fairness. This is because Facility-Location (the submodular function in these instances) models representation in contrast to diversity (modeled by LogDet), mining representative anchors, positives and negatives in SHASAM-MINE while learning representative features in SHASAM-LEARN.

8.5. Ablation: Compute Cost and Wall Clock time

We point out that introduction of the learning formulation in SHASAM does not add any additional parameters to the model. Nevertheless, particular instances of SHASAM like LogDetCMI requires computation of large matrices which scale with the increase in batch size requiring large compute infrastructure. We have discussed this in the limitations of our paper in Sec. 9.

Additionally, we also compare the wall-clock time requirements of SHASAM against SoTA methods in Fig. 1. For each setting in Fig. 1 we calculate the average time in phase 1 (phase 2 remains largely unchanged) across three distinct settings in CelebA. Our observations closely follow the discussion in Sec. 4.2 (main paper) which shows that adopting a combinatorial approach in SHASAM requires fewer training epochs in stage 1 (100 epochs in FSCL to $\sim$40 epochs in SHASAM). However, the selection of hard positives and negatives do add a computation overhead which reflects in the wall-clock times shown in Fig. 1. This has been listed as a limitation in Sec. 9.

8.6. Characterization of SHASAM-MINE on Synthetic Data

To characterize the effectiveness of the introduced SHASAM-MINE we simulate three different scenarios. These include variation in sample sizes between target attributes inducing *imbalance* and simulating *feature overlap* between groups inducing inter-group bias. Our goal is to show that minimizing L_{SHASAM} on the mined anchors, hard positives and negatives facilitate the learning of strong decision boundaries between target attributes while ensuring learning of compact feature clusters (minimize intra-group variance) for each target attribute label without biasing on the sensitive attribute labels.

Balanced Settings In this case both clusters A and B as shown in Fig. 3 have equal number of samples with distinguishable decision boundary between them. In contrast to *Random* selection which is the most widely used technique, SHASAM-MINE selects diverse anchors representing the complete target set (in this case cluster B). Additionally, our approach selects hard-positives which lie at the cluster boundary of B. Minimizing the separation between these hard-positives and anchors result in minimization of intra-group variance encouraging the model to be unbiased to the sensitive attributes. Lastly, hard-negatives in A also lie at the cluster boundary between A and B, resulting in increased decision boundary when the separation between hard-positives and and negatives are maximized through L_{SHASAM}.

Imbalanced Settings In this case clusters A and B as shown in Fig. 4 demonstrate imbalance with A being the rare group, with distinguishable decision boundary between them. Alongside selection of diverse anchors, SHASAM-MINE selects hard-positives and negatives which continue to lie at the cluster boundary of A and B irrespective of the choice of target attribute label (anchors are mined from this set) - Rare class A in Fig. 4 and abundant class B in

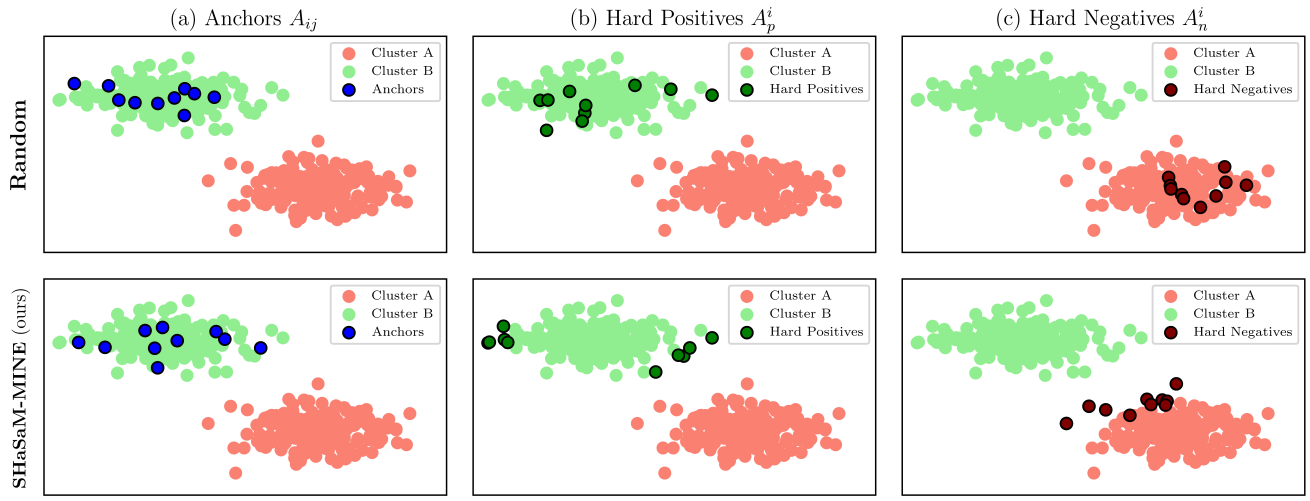

Figure 3. **Contrasting Random and SHASAM-MINE selection strategies** on a synthetic two-cluster imbalanced dataset to identify (a) Anchors, (b) Hard Positives and (c) Hard Negatives, in the *balanced* setting. The dataset generation and sample selection in performed under the same seed.

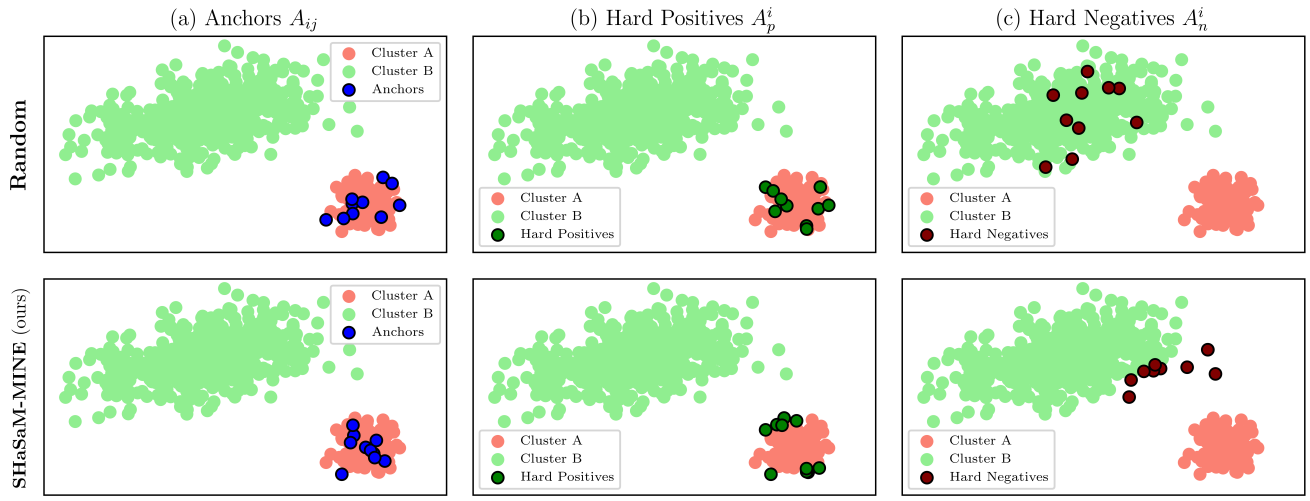

Figure 4. **Contrasting Random and SHASAM-MINE selection strategies** on a synthetic two-cluster imbalanced dataset with target attribute as the minority (rare) class. SHASAM-MINE identifies (a) Anchors, (b) Hard Positives and (c) Hard Negatives, showing the effectiveness of SHASAM in modeling the decision boundary between target attributes. The dataset generation and sample selection in performed under the same seed.

Fig. 4 of main paper. Minimizing L_{SHASAM} over the mined sets promotes reduction in intra-group variance within the groups and maximizes inter-group separation between the target attribute and the remaining groups encouraging the model to learn features, unbiased to the sensitive attributes.

Feature Similarity (Cluster Overlap) In contrast to distinct cluster boundaries depicted in Fig. 3 and Fig. 4 we introduce a case with high inter-group bias demonstrated as overlapping feature clusters A and B in an imbalanced setting (B being the abundant and A being the rare group). The behavior of SHASAM-MINE is consistent with the previous setting and mines hard positives and negatives at the cluster boundary. Interestingly, we see that SHASAM-MINE selects hard-negatives that largely represent the overlapped section of the embedding space which when consumed by L_{SHASAM} would promote mitigation of inter-group bias and enforce strong decision boundaries between target attributes.

8.7. Standard Deviation of Results on CelebA

We indicate in Section 4.1 of the main paper that we report the average performance over three independent runs by

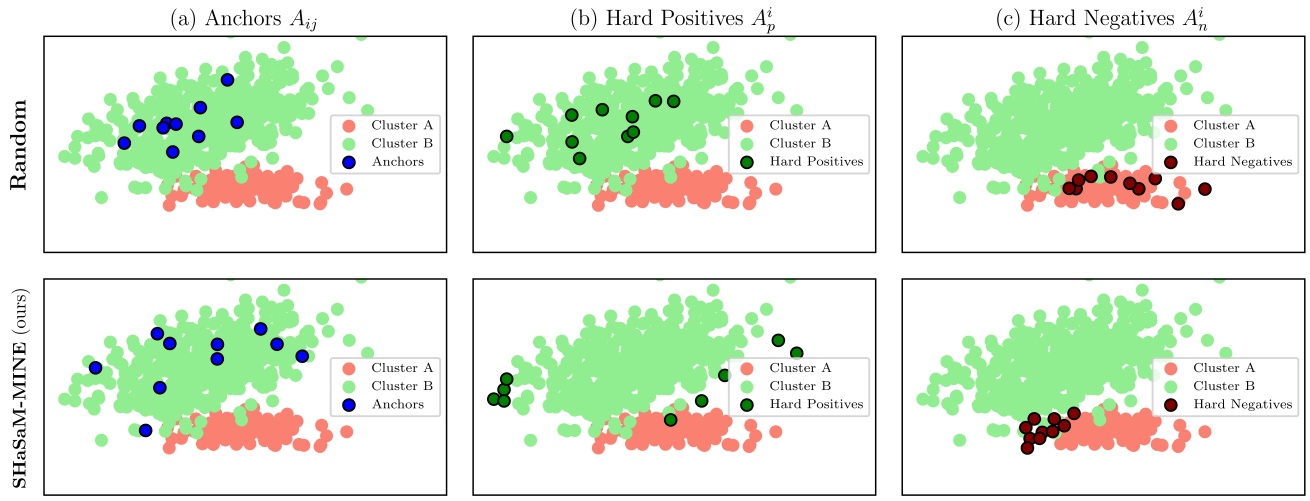

Figure 5. **Contrasting Random and SHASAM-MINE selection strategies** on a synthetic two-cluster imbalanced dataset to identify (a) Anchors, (b) Hard Positives and (c) Hard Negatives, when large feature similarity exists between clusters (overlapping clusters). The dataset generation and sample selection in performed under the same seed.

varying the seed value of underlying libraries among three random seeds. We supplement the results in Table 2 with standard deviation numbers in Tab. 4 for SHASAM variants and methods which were re-implemented by us. We show that models trained with SHASAM achieves the lowest standard deviations among all compared approaches. This can be attributed to the combinatorial formulation which allows the model to learn from complete sets of anchors, positives and negatives rather than contrasting individual samples [34] or one anchor pair and multiple positives and negatives [26].

9. Limitations

While the SHASAM framework presents a novel combinatorial approach for fair facial attribute recognition, it has a few limitations. The primary constraint is the computational cost and wall-clock time overhead associated with the SHASAM-MINE stage, which involves combining submodular optimization and representation learning in a unified framework. To remedy this, SHASAM relies on training with a randomly selected subset of the dataset at each epoch, which could be inefficient for extremely large-scale applications. Further, SHASAM operates on the assumption that discrete target and sensitive attributes are available, and in experiments, continuous attributes like 'age' were simplified into binary categories, a step that may not be suitable for all real-world scenarios. Finally, complex submodular functions like Log-Determinant may require computing large matrices (given a large batch size) which might expand the compute requirements beyond the available budget.

References

[1] Eric Baum and Frank Wilczek. Supervised learning of probability distributions by neural networks. In *Neural Information Processing Systems*, 1987. 1, 11

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *Intl. Conf. on Machine Learning (ICML)*, 2020. 1

[3] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 1

[4] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019. 1

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 3

[6] Nicholas Frosst, Nicolas Papernot, and Geoffrey E. Hinton. Analyzing and improving representations with the soft nearest neighbor loss. In *International Conference on Machine Learning*, 2019. 1

[7] Satoru Fujishige. *Submodular Functions and Optimization*. Elsevier, 2005. 1

[8] Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 2010. 1

Table 4. **Classification results with standard deviations on CelebA** measured in terms of Top-1 Accuracy (Acc.) and equalized odds (EO) by varying the target T and sensitive attributes S. Here, a, b, e, m, and y denote attractiveness, big nose, bags-under-eyes, male, and young, respectively. All results are averaged over three independent runs. * indicates our re-implementations. All values are rounded off to 1 decimal point.

Method	$T=a$ / $S=m$		$T=a$ / $S=y$		$T=b$ / $S=m$		$T=b$ / $S=y$		$T=e$ / $S=m$		$T=e$ / $S=y$		$T=e$ & b / $S=m$		$T=a$ / $S=m$ & y	
	EO (↓)	Acc. (↑)	EO (↓)	Acc. (↑)	EO (↓)	Acc. (↑)	EO (↓)	Acc. (↑)	EO (↓)	Acc. (↑)	EO (↓)	Acc. (↑)	EO (↓)	Acc. (↑)	EO (↓)	Acc. (↑)
CE [1]	27.8	79.6	16.8	79.8	17.6	84.0	14.7	84.5	15.0	83.9	12.7	83.8	12.9	72.6	31.3	79.5
GRL [27]	24.9	77.2	14.7	74.6	14.0	82.5	10.0	83.3	6.7	81.9	5.9	82.3	9.4	71.4	22.9	78.6
LNL [17]	21.8	79.9	13.7	74.3	10.7	82.3	6.8	82.3	5.0	81.6	3.3	80.3	7.4	70.8	20.7	77.7
FD-VAE [25]	15.1	76.9	14.8	77.5	11.2	81.6	6.7	81.7	5.7	82.6	6.2	84.0	8.2	70.2	19.9	78.0
MFD [12]	7.4	78.0	14.9	80.0	7.3	78.0	5.4	78.0	8.7	79.0	5.2	78.0	9.0	70.0	19.4	76.1
SupCon [15]	30.5	80.5	21.7	80.1	20.7	84.6	16.9	84.4	20.8	84.3	10.8	84.0	12.5	72.7	24.4	81.7
FSCL (w/ group norm) [26]*	6.5 ±0.4	79.1 ±0.4	12.4 ±0.5	79.1 ±0.5	4.7 ±0.5	82.9 ±0.4	4.8 ±0.3	84.1 ±0.5	3.0 ±0.4	83.4 ±0.6	**1.6** ±0.3	83.5 ±0.3	2.5 ±0.6	70.8 ±0.5	17.0 ±0.5	77.2 ±0.5
SHASAM(w/ LogDetCMI)	6.1 ±0.2	80.7 ±0.3	10.5 ±0.5	78.6 ±0.3	3.6 ±0.6	84.5 ±0.2	4.2 ±0.3	85.9 ±0.3	2.7 ±0.2	85.0 ±0.2	1.7 ±0.6	84.0 ±0.4	2.5 ±0.2	71.3 ±0.6	15.8 ±0.4	78.7 ±0.3
SHASAM(w/ FLCMI)	**5.5** ±0.3	**81.3** ±0.4	**9.9** ±0.7	**79.6** ±0.4	**3.3** ±0.6	**84.7** ±0.4	**3.9** ±0.5	**87.0** ±0.3	**2.6** ±0.3	**85.8** ±0.7	**1.6** ±0.4	**84.4** ±0.2	**2.2** ±0.4	**71.8** ±0.5	**14.6** ±0.3	**79.5** ±0.5
FSCL (w/ ViT) [26]	5.7 ±0.4	79.9 ±0.2	10.8 ±0.5	80.0 ±0.3	4.0 ±0.3	85.0 ±0.3	4.6 ±0.4	88.2 ±0.6	2.9 ±0.4	87.7 ±0.5	1.7 ±0.5	86.3 ±0.5	2.5 ±1.0	72.8 ±1.1	16.3 ±0.6	81.8 ±0.6
FairViT [34]*	6.4 ±0.2	83.8 ±0.4	12.6 ±0.2	82.5 ±0.1	5.1 ±0.4	86.1 ±0.1	4.7 ±0.2	89.3 ±0.1	3.4 ±0.3	88.1 ±0.3	1.8 ±0.3	**86.8** ±0.3	2.9 ±0.8	**74.4** ±0.9	17.4 ±0.5	82.4 ±0.5
SHASAM(w/ FLCMI, ViT)	**5.3** ±0.2	**84.0** ±0.3	**9.9** ±0.4	**82.3** ±0.1	**3.0** ±0.3	**85.6** ±0.2	**3.7** ±0.2	**89.2** ±0.2	**2.8** ±0.3	**88.2** ±0.3	**1.6** ±0.2	86.8 ±0.3	**2.2** ±0.7	74.3 ±1.0	14.6 ±0.6	**82.4** ±0.4

[9] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[10] Eeshaan Jain, Tushar Nandy, Gaurav Aggarwal, Ashish V. Tendulkar, Rishabh K Iyer, and Abir De. Efficient data subset selection to generalize training across models: Transductive and inductive networks. In *Thirty-seventh Conference on Neural Information Processing Systems (NeurIPS)*, 2023. 2

[11] Stefanie Jegelka and Jeff Bilmes. Submodularity beyond submodular energies: Coupling edges in graph cuts. In *CVPR 2011*, 2011. 1, 2

[12] Sangwon Jung, Donggyu Lee, Taeeon Park, and Taesup Moon. Fair feature distillation for visual recognition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2021. 3, 11

[13] V. Kaushal, R. Iyer, K. Doctor, A. Sahoo, P. Dubal, S. Kothawade, R. Mahadev, K. Dargan, and G. Ramakrishnan. Demystifying multi-faceted video summarization: Tradeoff between diversity, representation, coverage and importance. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 452–461, 2019. 1, 2

[14] Vishal Kaushal, Sandeep Subramanian, Suraj Kothawade, Rishabh Iyer, and Ganesh Ramakrishnan. A framework towards domain specific video summarization. In *2019 IEEE winter conference on applications of computer vision (WACV)*, pages 666–675. IEEE, 2019. 2

[15] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, 2020. 1, 3, 4, 11

[16] Krishnateja Killamsetty, Guttu Sai Abhishek, Aakriti, Ganesh Ramakrishnan, Alexandre V. Evfimievski, Lucian Popa, and Rishabh Iyer. AUTOMATA: gradient based data subset selection for compute-efficient hyper-parameter tuning. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, 2024. 2

[17] Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3, 11

[18] Suraj Kothawade, Vishal Kaushal, Ganesh Ramakrishnan, Jeff A. Bilmes, and Rishabh K. Iyer. PRISM: A rich class of parameterized submodular information measures for guided data subset selection. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI*, pages 10238–10246, 2022. 1, 2

[19] Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011. 1, 2

[20] Anay Majee, Suraj Nandkishor Kothawade, Krishnateja Killamsetty, and Rishabh K Iyer. SCoRe: Submodular combinatorial representation learning. In *Proceedings of the 41st International Conference on Machine Learning*, pages 34327–34349, 2024. 2, 7

[21] Anay Majee, Ryan Sharp, and Rishabh Iyer. Smile: Leveraging submodular mutual information for robust few-shot object detection. In *European Conference on Computer Vision (ECCV)*, 2024. 2

[22] Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, Amin Karbasi, Jan Vondrak, and Andreas Krause. Lazier than lazy greedy. *Proceedings of the AAAI Conference on Artificial Intelligence*, 29(1), 2015. 2

[23] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14(1):265–294, 1978. 2, 3, 7

[24] Patrik Okanovic, Roger Waleffe, Vasilis Mageirakos, Konstantinos Nikolakakis, Amin Karbasi, Dionysios Kalogerias, Nezihe Merve Gürel, and Theodoros Rekatsinas. Repeated random sampling for minimizing the time-to-accuracy of

learning. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[25] Sungho Park, Sunhee Hwang, Dohyung Kim, and Hyeran Byun. Learning disentangled representation for fair facial attribute classification via fairness-aware information alignment. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(3):2403–2411, 2021. 3, 11

[26] Sungho Park, Jewook Lee, Pilhyeon Lee, Sunhee Hwang, Dohyung Kim, and Hyeran Byun. Fair contrastive learning for facial attribute classification. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3, 4, 7, 8, 10, 11

[27] Edward Raff and Jared Sylvester. Gradient reversal against discrimination: A fair neural network learning approach. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, page 189–198. IEEE, 2018. 3, 11

[28] Vikram V. Ramaswamy, Sunnie S. Y. Kim, and Olga Russakovsky. Fair attribute classification through latent space de-biasing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9301–9310, 2021. 8

[29] Kanchana Ranasinghe, Muzammal Naseer, Munawar Hayat, Salman Khan, and Fahad Shahbaz Khan. Orthogonal projection loss. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 1

[30] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1

[31] Aili Shen, Xudong Han, Trevor Cohn, Timothy Baldwin, and Lea Frermann. Contrastive learning for fair representations. *ArXiv*, abs/2109.10645, 2021. 4

[32] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in Neural Inf. Processing Systems*, 2016. 1

[33] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 1

[34] Bowei Tian, Ruijie Du, and Yanning Shen. Fairvit: Fair vision transformer via adaptive masking, 2024. 3, 7, 8, 10, 11

[35] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 1

[36] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2019. 1