Deep Fair Clustering for Visual Learning
Supplementary Material

A. Proof

In this section we provide the detailed proof of Theorem 4.1 in the main paper. Before we give the proof, we first present a useful lemma that will be used during the proof.

Lemma A.1 (log-sum inequality)**.** Let $\{a_i\}_{i\in[n]}$ and $\{b_i\}_{i\in[n]}$ be two sequence of nonnegative numbers. Define $a := \sum_{i\in[n]} a_i$ and $b := \sum_{i\in[n]} b_i$, then the following inequality holds:

$$a \log \frac{a}{b} \leq \sum_{i\in[n]} a_i \log \frac{a_i}{b_i}, \tag{11}$$

where the equality holds iff a_i/b_i are equal for all $i \in [n]$.

Now we are ready to prove Theorem 4.1.

Theorem 4.1. Let $X \sim \mathcal{D}$ be the input random variable of the clustering algorithm C. If the clustering assignment $C(X)$ is independent of the sensitive attribute G, then

$$I(Y; C(X)) \leq \sum_{g\in[M]} \Pr(G = g) \cdot I(Y_g; C(X_g)), \tag{9}$$

where X_g is the input random variable in the g-th protected subgroup and Y_g is the corresponding external label of X_g.

Proof. To simplify the notation, in what follows we use $C = C(X)$ and drop the subscript $\mathcal{D}$ from the symbol $\Pr_{\mathcal{D}}(\cdot)$. By definition of the mutual information, we have:

$$\begin{aligned} I(Y; C(X)) &= \sum_{ij} \Pr(Y = i, C = j) \log \frac{\Pr(Y = i, C = j)}{\Pr(Y = i) \cdot \Pr(C = j)} \\ &= \sum_{ij} \big(\sum_{g\in[M]} \Pr(Y = i, C = j, G = g) \big) \log \frac{\sum_{g\in[M]} \Pr(Y = i, C = j, G = g)}{\big(\sum_{g\in[M]} \Pr(Y = i, G = g) \big) \cdot \Pr(C = j)} \end{aligned}$$

To make the presentation uncluttered, we define $p_{ijg} := \Pr(Y = i, C = j, G = g)$, $p_i := \Pr(Y = i)$, $p_j := \Pr(C = j)$ and $p_g := \Pr(G = g)$.

$$= \sum_{ij} \big(\sum_{g} p_{ijg} \big) \log \frac{\sum_g p_{ig|j}}{\sum_g p_{ig}}. \tag{12}$$

Now consider a fixed pair of i, j, by the log-sum inequality in Lemma A.1, we can upper bound $\big(\sum_g p_{ijg} \big) \log \frac{\sum_g p_{ig|j}}{\sum_g p_{ig}}$ as

$$\big(\sum_g p_{ijg} \big) \log \frac{\sum_g p_{ig|j}}{\sum_g p_{ig}} = p_j \big(\sum_g p_{ig|j} \big) \log \frac{\sum_g p_{ig|j}}{\sum_g p_{ig}} \leq p_j \sum_g p_{ig|j} \log \frac{p_{ig|j}}{p_{ig}} = \sum_g p_g p_{ij|g} \log \frac{p_{ijg}}{p_{ig} \cdot p_j}.$$

Furthermore, since $C(X)$ is independent of G, then $\forall g \in [M]$, the following equality holds:

$$\sum_g p_g p_{ij|g} \log \frac{p_{ijg}}{p_{ig} \cdot p_j} = \sum_g p_g p_{ij|g} \log \frac{p_{ij|g}}{p_{i|g} \cdot p_j} = \sum_g p_g p_{ij|g} \log \frac{p_{ij|g}}{p_{i|g} \cdot p_{j|g}},$$

where in the last equality we use $p_j = p_{j|g}$ since $C(X) = j$ is independent of $G = g$. Now substituting the above inequality back to (12) yields:

$$\begin{aligned}\sum_{ij} \big(\sum_g p_{ijg}\big) \log \frac{\sum_g p_{ig|j}}{\sum_g p_{ig}} &\leq \sum_{ij} \sum_g p_g p_{ij|g} \log \frac{p_{ij|g}}{p_{i|g} \cdot p_{j|g}} \\ &= \sum_g p_g \big(\sum_{ij} p_{ij|g} \log \frac{p_{ij|g}}{p_{i|g} \cdot p_{j|g}} \big) \\ &= \sum_g \Pr(G = g) \cdot I(Y_g; C(X_g)),\end{aligned}$$

which completes the proof. □

Proposition 4.1. Let $X \sim \mathcal{D}$ be the input random variable of the clustering algorithm C. If the clustering assignment $C(X)$ is independent of the sensitive attribute G, then

$$H(C(X)) = \sum_{g \in [M]} \Pr(G = g) \cdot H(C(X_g)). \quad (10)$$

Proof. Under the assumption that $C(X)$ is independent of G and by definition of conditional entropy, we have:

$$H(C(X)) = H(C(X) \mid G) = \sum_{g \in [M]} \Pr(G = g) \cdot H(C(X) \mid G = g) = \sum_{g \in [M]} \Pr(G = g) \cdot H(C(X_g)),$$

completing the proof. □

B. Label Matching

In this section we provide the approximate calculation for clustering accuracy of the best and worst matching on each dataset. We firstly show the clustering accuracy we get on each protected subgroup in Table 5.

Table 5. Clustering accuracy on each protected subgroup.

Subgroup	*MNIST-USPS* / MNIST	*MNIST-USPS* / USPS	*Color Reverse MNIST* / Original	*Color Reverse MNIST* / Reversed
Accuracy	0.894	0.745	0.929	0.965
Subgroup	*MTFL* / With Glasses	*MTFL* / Without Glasses	*Office-31 / Amazon*	*Office-31 / Webcam*
Accuracy	0.633	0.760	0.690	0.823

To simplify the matching calculation, we assume that each cluster contains equivalent numbers of samples and have equal accuracy. We denote the size of a protected subgroup m as s_m and the accuracy as Acc_m. In *MNIST-USPS*, *Color Reverse MNIST*, and *Office-31*, the number of cluster K is large, and the best and worst matching accuracy can be calculated as:

$$Acc_{best} = \sum_{m \in [M]} \frac{s_m \cdot Acc_m}{\sum_{m \in [M]} s_m}, \quad Acc_{worst} = \frac{max\{s_m \cdot Acc_m\}_{m=1}^M}{\sum_{m \in [M]} s_m}. \quad (13)$$

However, when $K = 2$ and $M = 2$ in *MTFL* dataset, consider the final accuracy will be significantly contributed by the inconsistent samples within one cluster, we have the accuracy of worst matching for protected subgroup m and m' as:

$$Acc_{worst} = \frac{max\{s_m \cdot Acc_m + s_{m'} \cdot (1 - Acc_{m'}),\ s_m \cdot (1 - Acc_m) + s_{m'} \cdot Acc_{m'}\}}{\sum_{m \in [M]} s_m}. \quad (14)$$