Attention IoU: Examining Biases in CelebA using Attention Maps

Supplementary Material

A. Gradients for GradCAM

In Sec. 3.1, to compute GradCAM for image features that contribute positively, we describe taking the gradient of the absolute value of the class output $|y_a|$ for binary cross-entropy loss, while taking gradient of class output y_a directly for categorical cross-entropy loss.

When using a model that is trained using binary cross-entropy loss, computing the gradient w.r.t. the absolute value of the logit (before the sigmoid) is equivalent to computing the gradient w.r.t. to the predicted class for categorical cross-entropy loss with two heads (one each for the positive and negative class). Concretely, let s be the value of the logit; the probability that this model assigns to the positive class is $\sigma(s) = \frac{1}{1+e^{-s}}$, and the probability assigned to the negative class is $1 - \sigma(s) = \frac{e^{-s}}{1+e^{-s}} = \sigma(-s)$. The model prediction is $\arg\max(\sigma(s), \sigma(-s)) = \arg\max(s, -s)$. Thus, taking the gradient with respect to the absolute value of the logits allows us to find positive contributions to the predicted binary class.

B. Proofs of Invariants

In Sec. 3.2, we introduce the Attention-IoU metric, $\mathcal{B}_{\text{A-IoU}}$, which is invariant to scale and size for pixel maps.

First, we confirm that if the two input maps are identical, $\mathbf{M}_1 = \mathbf{M}_2 = \mathbf{M} \in \mathbb{R}^{h \times w}$, the Attention-IoU metric is 1:

$$\mathcal{B}_{\text{A-IoU}}(\mathbf{M}, \mathbf{M}) = \frac{\langle \widehat{\mathbf{M}}, \widehat{\mathbf{M}} \rangle_F}{\left\| \frac{\widehat{\mathbf{M}} + \widehat{\mathbf{M}}}{2} \right\|_F^2} \tag{4}$$

$$= \frac{\langle \widehat{\mathbf{M}}, \widehat{\mathbf{M}} \rangle_F}{\left\| \widehat{\mathbf{M}} \right\|_F^2} = \frac{\left\| \widehat{\mathbf{M}} \right\|_F^2}{\left\| \widehat{\mathbf{M}} \right\|_F^2} = 1. \tag{5}$$

We next prove that $\mathcal{B}_{\text{A-IoU}}$ is scale invariant. Given two maps $\mathbf{M}_1, \mathbf{M}_2 \in \mathbb{R}^{h \times w}$, suppose the maps are multiplied by the scalars $a_1, a_2 \in \mathbb{R}_+$ respectively. Then their L_1 normalized maps are

$$\widehat{a_i \mathbf{M}_i} = \frac{a_i \mathbf{M}_i}{||a_i \mathbf{M}_i||_1} = \frac{a_i \mathbf{M}_i}{a_i ||\mathbf{M}_i||_1} = \widehat{\mathbf{M}}_i \tag{6}$$

So $\mathcal{B}_{\text{A-IoU}}(a_1 \mathbf{M}_1, a_2 \mathbf{M}_2) = \mathcal{B}_{\text{A-IoU}}(\mathbf{M}_1, \mathbf{M}_2)$.

For the proof of size invariance, we assume for simplicity that the maps are resized by a positive integer scalar $\alpha \in \mathbb{N}$ using nearest neighbor interpolation. Again, consider two maps $\mathbf{M}_1, \mathbf{M}_2 \in \mathbb{R}^{h \times w}$. Let $\mathbf{M}_1^\alpha, \mathbf{M}_2^\alpha \in \mathbb{R}^{\alpha h \times \alpha w}$ be the rescaling of the two maps by the constant α. For example, with $\alpha = 2$, a 5×5 box in the center of the map will be resized to be a 10×10 box, with the same spacial location within the map. Note that the L_1 normalized maps are

$$\widehat{\mathbf{M}}_i^\alpha = \frac{\mathbf{M}_i^\alpha}{||\mathbf{M}_i^\alpha||_1} = \frac{\mathbf{M}_i^\alpha}{\alpha^2 ||\mathbf{M}_i||_1}, \tag{7}$$

as each pixel in the original map appears α^2 times in the resized map. Furthermore, the Frobenius inner product of the two resized maps is

$$\langle \mathbf{M}_1^\alpha, \mathbf{M}_2^\alpha \rangle_F = \sum_{i=1}^{\alpha h} \sum_{j=1}^{\alpha w} (\mathbf{M}_1^\alpha)_{ij} \cdot (\mathbf{M}_2^\alpha)_{ij} \tag{8}$$

$$= \alpha^2 \sum_{i=1}^{h} \sum_{j=1}^{w} (\mathbf{M}_1)_{ij} \cdot (\mathbf{M}_2)_{ij} \tag{9}$$

$$= \alpha^2 \langle \mathbf{M}_1, \mathbf{M}_2 \rangle_F \tag{10}$$

and, for the norm,

$$\left\| \frac{\widehat{\mathbf{M}}_1^\alpha + \widehat{\mathbf{M}}_2^\alpha}{2} \right\|_F^2 = \frac{1}{4} \sum_{i=1}^{\alpha h} \sum_{j=1}^{\alpha w} \left(\frac{(\mathbf{M}_1^\alpha)_{ij}}{||\mathbf{M}_1^\alpha||_1} + \frac{(\mathbf{M}_2^\alpha)_{ij}}{||\mathbf{M}_2^\alpha||_1} \right)^2 \tag{11}$$

$$= \frac{1}{4\alpha^4} \sum_{i=1}^{\alpha h} \sum_{j=1}^{\alpha w} \left(\frac{(\mathbf{M}_1^\alpha)_{ij}}{||\mathbf{M}_1||_1} + \frac{(\mathbf{M}_2^\alpha)_{ij}}{||\mathbf{M}_2||_1} \right)^2 \tag{12}$$

$$= \frac{1}{4\alpha^2} \sum_{i=1}^{h} \sum_{j=1}^{w} \left(\frac{(\mathbf{M}_1)_{ij}}{||\mathbf{M}_1||_1} + \frac{(\mathbf{M}_2)_{ij}}{||\mathbf{M}_2||_1} \right)^2 \tag{13}$$

$$= \frac{1}{\alpha^2} \left\| \frac{\widehat{\mathbf{M}}_1 + \widehat{\mathbf{M}}_2}{2} \right\|_F^2. \tag{14}$$

Thus, combining the two parts together,

$$\mathcal{B}_{\text{A-IoU}}(\mathbf{M}_1^\alpha, \mathbf{M}_2^\alpha) = \frac{\langle \widehat{\mathbf{M}}_1^\alpha, \widehat{\mathbf{M}}_2^\alpha \rangle_F}{\left\| \frac{\widehat{\mathbf{M}}_1^\alpha + \widehat{\mathbf{M}}_2^\alpha}{2} \right\|_F^2} \tag{15}$$

$$= \frac{\frac{1}{\alpha^4 ||\mathbf{M}_1||_1 \cdot ||\mathbf{M}_2||_1} \langle \mathbf{M}_1^\alpha, \mathbf{M}_2^\alpha \rangle_F}{\frac{1}{\alpha^2} \left\| \frac{\widehat{\mathbf{M}}_1 + \widehat{\mathbf{M}}_2}{2} \right\|_F^2} \tag{16}$$

$$= \frac{\frac{1}{\alpha^2} \frac{1}{||\mathbf{M}_1||_1 \cdot ||\mathbf{M}_2||_1} \langle \mathbf{M}_1, \mathbf{M}_2 \rangle_F}{\frac{1}{\alpha^2} \left\| \frac{\widehat{\mathbf{M}}_1 + \widehat{\mathbf{M}}_2}{2} \right\|_F^2} \tag{17}$$

$$= \frac{\langle \widehat{\mathbf{M}}_1, \widehat{\mathbf{M}}_2 \rangle_F}{\left\| \frac{\widehat{\mathbf{M}}_1 + \widehat{\mathbf{M}}_2}{2} \right\|_F^2} \tag{18}$$

$$= \mathcal{B}_{\text{A-IoU}}(\mathbf{M}_1, \mathbf{M}_2). \tag{19}$$

Although in the proof $\mathbf{M}_1^\alpha$ and $\mathbf{M}_2^\alpha$ are larger matrices than $\mathbf{M}_1$ and $\mathbf{M}_2$, the same argument applies if $\mathbf{M}_1$ and $\mathbf{M}_2$ are zero-padded to have same dimensions as the resized maps.

C. Subsampling Training Details

Here we provide experimental details for varying training set correlations in Sec. 5.2. Given a target Matthews correlation coefficient between the specified attribute and `Male`, we find subgroup sizes that achieve the target MCC (as MCC is dependent entirely on the sizes of the 4 subgroups) using SciPy's `optimize.minimize` with the trust region method[2] (Fig. 10). We bound the sizes of the subsampled subgroups to the size of the original groups, and aim to minimize the distance to the original group sizes by the L_2 norm. To reduce fluctuations between the subsampled sizes, we initialize the optimizer with the adjacent subgroup sizes, with the original subgroups sizes in the training set as the starting point. Lastly, after running the optimization once for all MCCs, we rerun the optimization process with the additional bound of the smallest subsampled training set, so that all the subsampled training sets are of the same size. As the subsampling was an ablation study, the heatmap scores reported in Fig. 9 were run on the validation set.

D. Additional CelebA Results

Model Evaluation. The average precision weighted for all 40 attributes in CelebA, averaged across the 20 trained models with the experimental setup detailed in Sec. 5.1, is 0.902 ± 0.025. For reference, the normalized average precision (AP_N) [25] for the `Male` attribute is 0.994 ± 0.003, the second highest after `Eyeglasses` (0.998 ± 0.001). In Fig. 11 we show average heatmaps for select attributes.

[2] https://docs.scipy.org/doc/scipy/reference/optimize.minimize-trustconstr.html

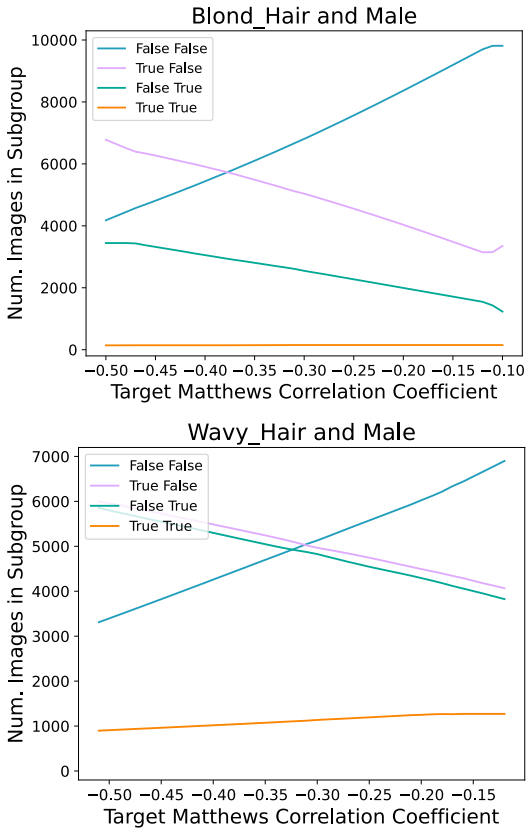

Figure 10. **Training set subgroup sizes under subsampling**. Here we report subgroup sizes of the training set of varying MCCs for `Blond_Hair` and `Wavy_Hair` with `Male`, under our optimization scheme, to compute the results in Sec. 5.2 and Fig. 9. Subgroup sizes are bounded to the smallest subsampled training set size. The legend shows the four different subgroups groups, with the first value indicating the target label and the second `Male`.

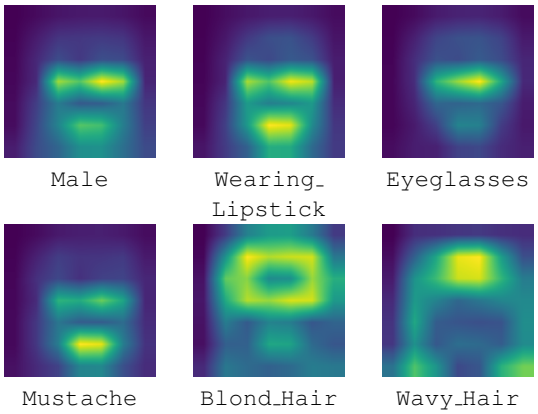

Figure 11. **Average heatmaps for CelebA attributes**. We visualize average heatmaps for the selected attributes investigated in Sec. 5.2.

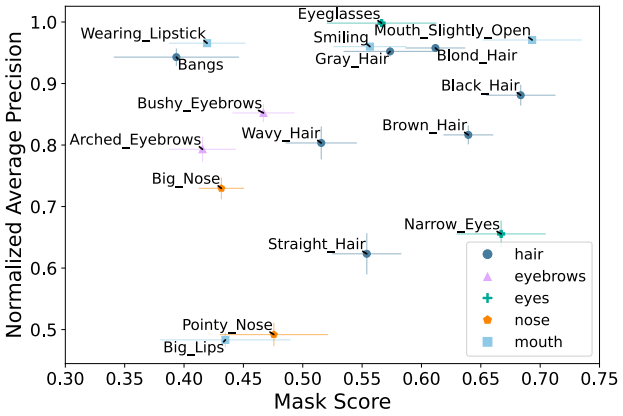

Figure 12. **Evaluation of mask score using GradCAM on CelebA test set with attribute-specific feature masks, compared to average precision**. To compare per-attribute AP between attributes, we adopt Hoiem *et al.*'s normalized average precision (AP_N) metric [25].

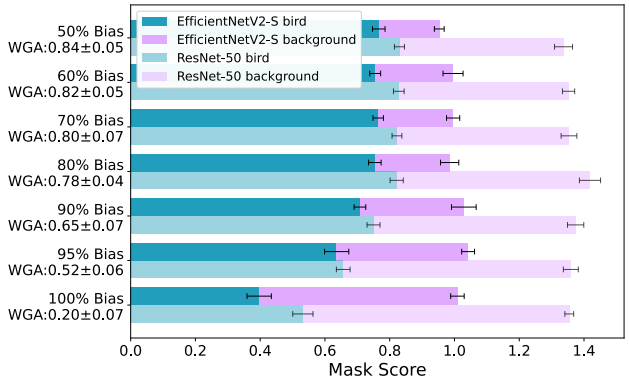

Figure 13. **EfficientNetV2 mask score on Waterbirds**. The top bars indicate Attention-IoU mask scores for EfficientNetV2-S models, while the bottom bars are corresponding ResNet-50 scores from Fig. 3. WGA is for the EfficientNet model. As with ResNet, the EfficientNet models attend less to the bird and more to the background, mirroring the decrease in WGA.

CelebA Normalized Average Precision. As a comparison to Fig. 5, which shows CelebA mask score against worst group accuracy, in Fig. 12 we show the mask score of the same 17 attributes to their normalized average precision (AP_N). Compared with worst group accuracy, there is a no correlation for normalized average precision with respect to the mask score. Unlike worst group accuracy, to calculate normalized average precision one does not need to assume the correlated attribute.

E. Evaluating with EfficientNet

To demonstrate the effectiveness of Attention-IoU on architectures other than ResNet, we also evaluated the metric using the EfficientNetV2-S architecture [69] on both the Waterbirds and CelebA datasets. Aside from the change in architecture, and averaging over 10 trained models instead of 20, the experimental setup remained the same.

For Waterbirds, the EfficientNet models show a very similar pattern to ResNet in attending less to the bird and more to the background as dataset bias increases (Fig. 14). The EfficientNet heatmap scores for CelebA also show a strong positive trend with MCC like ResNet (Fig. 13). The 5 highlighted attributes maintain their relative positions, with some changes owing to different architectures and pretraining weights.

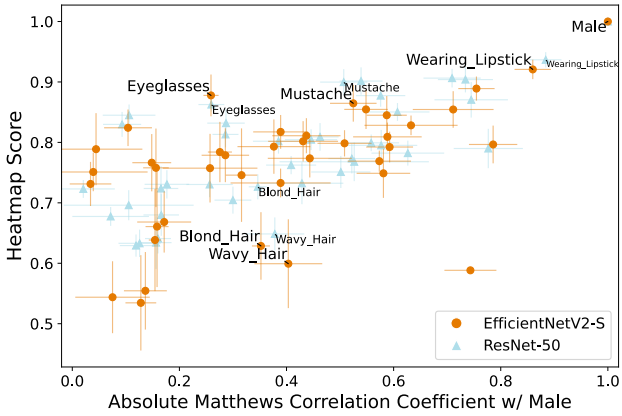

Figure 14. **EfficientNetV2 heatmap scores on CelebA attributes.** Orange/circle indicates results with EfficientNetV2-S models, and light blue/triangle are ResNet-50 results from Fig. 5. We observe a very similar trend in EfficientNetV2 to that of ResNet-50. Highlighted attributes maintain their relative position, with some movement owing to different architectures and pretraining weights.