COMPUTER VISION (2024-2025)

REPORT: **Task 2**

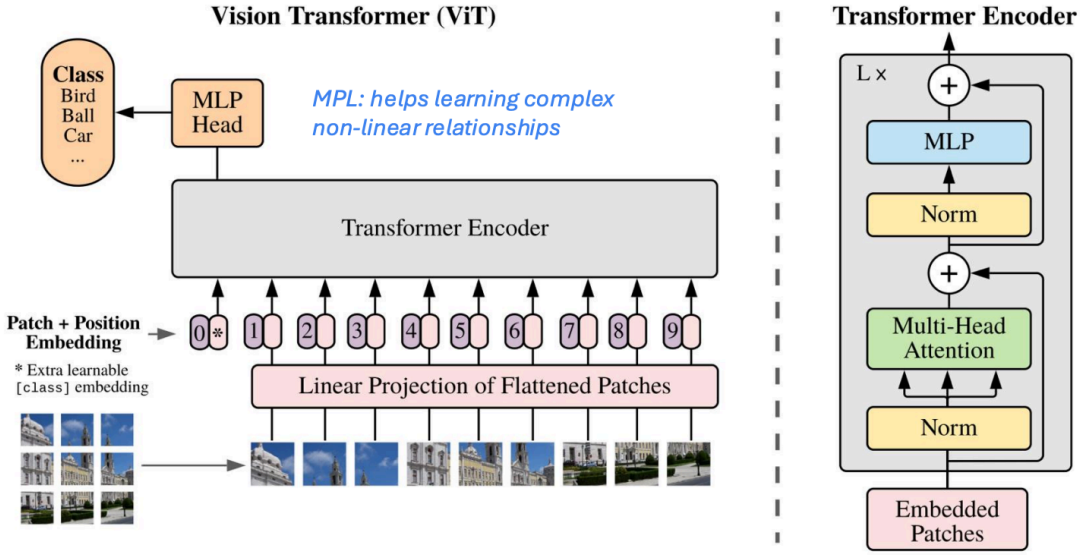
Group members:

Olmo Gordon Rodriguez
Iris Vukovic

1. MODEL DESCRIPTION

During task 1, we adapted the given baseline model i.e. ResNet-50 by adding some extra fully connected layers and combining dropout. This however did not result in any accuracy or bias improvements and in fact worsened bias results by a lot, as seen in Table 2.

We therefore opted for an implementation of a different model instead, the ViT pytorch model. This model was trained on 100% of train data, compared to ResNet-50 which was only trained on 70% of the train data, which we assumed would boost its accuracy. ViT was also pretrained on ImageNet, which is a larger database than Faces which ResNet-50 was pretrained on.



We altered the baseline ViT model to have a regression head instead of a classification head, since we needed it for the regression problem of predicting age.

2. BIAS MITIGATION STRATEGY

For our bias mitigation strategy in task 2 we constructed a multi-attribute weighted loss function that accounts for gender, ethnicity and facial expression. in the dataset. This approach ensures the model learns more effectively from underrepresented subgroups, focusing on reducing bias and improving overall accuracy.

The motivation behind our proposed solution stems from the realization that conventional loss functions do not sufficiently penalize errors for minority groups. Many existing bias mitigation techniques focus on a single attribute, such as age, but fail to incorporate multiple

demographic and categorical features, as displayed in the given baseline custom loss function. By weighting samples based on multiple attributes simultaneously, we aim to create a more equitable learning process that improves fairness and generalization across all demographic groups.

We therefore considered all available metadata attributes: gender, ethnicity and facial expression. Each sample in the dataset is assigned a weight based on the frequency of its attribute. Underrepresented attributes receive higher weights, ensuring that these samples contribute more significantly to the model's learning process.

We implemented a custom weighted MSE loss where we calculated the standard MSE loss, as usually done for regression tasks, and then multiplied it by an array of sample weights for each of the metadata subgroups based on how often they appeared in the entire dataset. The final loss was the mean of those weights. Each subgroup's weight was calculated by inverting its frequency, so samples from a subgroup that appeared less frequently in the dataset would have a higher weight.

With more time, we would try a custom loss that incorporates the age group over 60 category too since we only focused on metadata in this strategy.

3. TRAINING STRATEGY

As mentioned, due to the poor performance of our model during task 1, we decided to implement PyTorch's Visual Transformer's model. We adapted the baseline implementation to run with the custom loss function explained above. We trained in 10 epochs, which is way less than the 40 we used to train ResNet-50. ViT converges faster due to its architecture of capturing images as a bunch of patches, it captures global dependencies earlier on in the training. We tried 20 epochs occasionally when running experiments, but early stopping was often triggered and the results were not significantly impacted by more epochs, so we found 10 epochs to be optimal. We used the Adam optimizer, a learning rate of 1e-5, and the custom loss function we described above.

4. EXPERIMENTS AND RESULTS

Experiment 1: ViT with data augmentation
Augmentations: Horizontal flip, rotations of 10 degrees, color jittering, and affine transformation. We applied the augmentations once per sample to the samples that met the required conditions of being part of an underrepresented subgroup.

Compared to the baseline ViT without augmentation (testing on the validations set), the MAE was lower, as was the ethnicity bias and the expression bias, though age bias and gender bias increased. With more time, we would try other augmentation combinations and also try randomly applying certain augmentations to certain minority samples instead of applying equally on all minority samples.

Experiment 2: ViT with custom loss

This experiment gave the **best MAE**, but didn't significantly improve any bias scores.

Experiment 3: ViT with custom loss and data augmentation
Our final experiment didn't result in any improvements compared to Experiment 2 or compared to the baseline ViT model and only slightly improved age bias and MAE compared to Experiment 1. This came as a surprise since we thought that the bias mitigating power of both strategies combined would improve our model the most.

TABLE 2: Comparing the results of the final model using custom loss vs. some baseline models **on the Test Set**. Better results are highlighted in bold.

	Data aug.	**Custom Loss**	**Gender (bias)**	**Expression (bias)**	**Ethnicity (bias)**	**Age (bias)**	**Avg bias**	**MAE**
a) Starting-kit (ResNet-50)	NO	NO	0.1540	**0.1023**	**0.3441**	3.1021	0.9256	4.8119
b) Starting-kit (ResNet-50)	YES	NO	**0.0102**	0.3210	0.5005	**2.4936**	**0.8313**	**4.7332**
c) Extra 10 layers 2 stage (ResNet-50)	YES	NO	4.0059	2.4586	4.5119	33.493	11.1174	9.8547
d) Extra 10 layers 1 stage (ResNet-50)	YES	NO	4.007	2.4567	4.5014	33.4176	11.0957	7.76556
e) Extra 5 layers 2 stage (ResNet-50)	YES	NO	3.9873	2.4686	4.5033	33.4654	11.1061	15.84
f) ViT w/regression head	NO	NO	**0.0493**	0.2874	0.1837	**1.8804**	**0.6002**	4.2573
g) ViT w/regression head	YES	NO	0.2516	**0.1227**	**0.1067**	2.231	0.678	3.8384
h) ViT w/regression head	NO	YES	0.1916	0.1794	0.2571	2.0018	0.6574	**3.7721**
i) ViT w/regression head	YES	YES	0.3851	0.2356	0.3571	2.0841	0.7654	3.8335

5. FINAL REMARKS

In the case of ResNet-50, the strategies for bias reduction that we tried significantly worsened the model, but we began to see improvements when we started working with the Vision Transformer. We assumed that combining two bias mitigators would give us the best results, however that was not the case as seen in Table 2. With more time and computational power, we would experiment with different kinds of custom loss and combine them with different augmentation combinations. Once again, a balanced dataset from the start would be the best antidote to model bias.