COMPUTER VISION (2024-2025)

REPORT: **Task 1**

Group members:

Olmo Gordon Rodriguez
Iris Vukovic

1. MODEL DESCRIPTION

The final model is based on a pre-trained ResNet50 (VGGFace2) architecture, with additional fully connected layers that specialize in age estimation. The base model extracts high-level facial features, while the newly added layers are meant to refine these features for improved age prediction accuracy and bias mitigation. The modifications consist of 10 extra layers, which should progressively distill and regularize the extracted facial features before producing the final age estimate. These include:

- Three fully connected (dense) layers of decreasing size (512 → 256 → 128 units), each followed by:
    - Batch Normalization to stabilize feature distributions.
    - Dropout (50%, 30%, and 30% respectively) to reduce overfitting and ensure better generalization.
- A final linear output layer that predicts age as a continuous value.

This repeated pattern (dense → batch normalization → dropout) is applied three times, progressively reducing the feature space before the final prediction. The motivation behind this design is twofold:

1. Improving Accuracy:
    - The additional fully connected layers transform facial features into an age-sensitive feature space, improving the model's ability to capture non-linear relationships between facial attributes and age.
    - Progressive dimensionality reduction (512 → 256 → 128 units) forces the network to extract the most important aging cues while reducing noise.
    - Batch Normalization stabilizes activations and accelerates convergence, ensuring smoother and more efficient training.
2. Reducing Bias:
    - Dropout regularization prevents the model from overfitting to dominant demographic groups, improving its generalization across different subpopulations.
    - Batch Normalization helps normalize feature distributions, reducing sensitivity to biases in ethnicity, age, and gender distribution.
    - Using a pre-trained VGGFace2 model as a feature extractor ensures that the base facial features are already well-distributed across diverse ethnicities, minimizing demographic bias from the start.

2. BIAS MITIGATION STRATEGY

To mitigate bias in the dataset, we explored various data augmentation strategies, ultimately focusing on age and ethnicity, as these attributes showed the most significant imbalances. The dataset exhibited a heavy skew toward younger individuals and white subjects, with very few images of individuals over 60 years old or from underrepresented ethnic groups. Without addressing this imbalance, the model risked being biased toward the majority groups, leading to inaccurate predictions for minority subgroups.

We initially applied augmentations uniformly to all samples from the underrepresented groups, which significantly increased their representation in the dataset. This approach aimed to balance the training distribution by increasing the number of images for older individuals and people of color. However, after evaluating model performance, we observed that this strategy led to over-representation of certain augmented features, potentially causing the model to overfit to augmentation artifacts rather than meaningful age-related or ethnicity-related patterns.

To address this, we adjusted our augmentation strategy by randomly applying transformations to the minority subgroups rather than augmenting all their images equally. In order to further balance the ethnic subgroups, we also removed 30% of the caucasian samples. This approach increased diversity in the augmented dataset without drastically altering the sample size.

1. **Horizontal Flipping** – Provided additional diversity for non-asymmetrical features.
2. **Brightness and Contrast Adjustments** – Accounted for different lighting conditions.
3. **Gaussian Blur** – Simulated slight variations in camera focus and image quality.
4. **Rotation (≤ 15°)** – Introduced variability in pose while maintaining realistic appearances.

Gender and expression were not directly modified, as their distributions were more balanced in the dataset.

3. TRAINING STRATEGY

We experimented with multiple training strategies in one and two stages:

- Single stage training of extra 10 layers directly on augmented data
- Two-stage training of extra 10 layers on base data and full model on augmented data
- Two-stage training of baseline model on base data and full model on augmented data

Before our experiments, we believed that the two-stage training strategy with extra 10 layers would provide the best balance between accuracy, computational efficiency, and fairness because it would allow the new layers to learn specialized age-related features before fine-tuning the entire network.

In the two-stage approach with ten additional layers:

- We first trained only the 10 additional layers on top of the frozen ResNet50 base using the initial dataset to allow the new layers to specialize in age prediction without disrupting the pre-trained feature extraction.
- Then, we unfreeze all layers of the ResNet50 base to fine-tune the entire network for optimal performance and train it on the augmented data.

We therefore leveraged transfer learning by using a ResNet50 model pre-trained on VGGFace2 rather than training from scratch. This significantly reduced training time and allowed us to benefit from a robust facial feature extractor trained on a large-scale dataset.

We optimized our hyperparameters to balance efficiency and performance, stabilizing training and managing GPU constraints effectively:

- **Optimizer:** Adam – selected for its adaptive learning rate capabilities, which helped with stability in the two-stage training.
- **Batch Size:** Reduced from 32 to 16 to manage GPU memory constraints.
- **Epochs:** Reduced from 50 to 40 to improve efficiency while maintaining accuracy.
- **Learning Rate:** Increased from 1e-5 to 1e-4 to speed up convergence during initial training.
- **Early Stopping:** Applied with patience = 10 to prevent overfitting and unnecessary GPU usage.
- **Best Model Checkpointing:** Enabled to save the best-performing model based on validation loss.

Although none of our approaches improved the accuracy and bias of the baselines, we found that, in fact, the one stage training approach with ten extra layers actually produced slightly better results and had higher computational efficiency than the two stage approach.

4. EXPERIMENTS AND RESULTS

Overview:

At first, we trained the model in separate experiments with different augmentation strategies, one with the objective of reducing age bias and the other reducing ethnicity bias. Ultimately, we decided to train a model with the objective of simultaneously reducing the bias score of both ethnicity and age and ran separate experiments based on different training strategies instead of different augmentation strategies.

Initially, we applied the augmentations equally to all samples from the minority subgroups (people over the age of 60 and Asian and African-American ethnicity groups), and then transitioned to randomizing the transformations to avoid too much augmentation. We tried performing the augmentations various amounts of times ranging for one time to three times on the African American ethnicity group, since they had the smallest sample size. On every other group we applied one round of augmentations. In addition to augmenting minority groups, we also deleted some samples from the Caucasian group in an attempt to balance out the ethnicity distribution, since the Caucasian sample size was so much larger than both other groups. Our chosen augmentations were: horizontally flipping the image, rotating 15 degrees, gaussian blur, and reducing brightness, as detailed above.

We tried training in two stages, first stage on the original dataset and second stage on the augmented dataset. For the first stage, we experimented with downloading the pre-trained model, training the given model ourselves, and training a model with ten extra layers (as detailed above) ourselves. The downloaded pre-trained model had better results than when we trained the original model ourselves, but the model architecture we implemented with extra layers was best. We also tried training everything in one stage with extra layers on the augmented data and ended up getting comparable results from all the training strategies. We thought that two stages would be better because transfer learning, training on the full dataset in the first round and then fine-tuning in the second round, makes the model more robust to variations in the dataset. We ended up getting slightly better results with the one stage training strategy.

Goal: Reduce age and ethnicity bias

Experiment A:
Description: Two-stage training with both ethnicity and age augmentations (specific augmentations listed above) applied randomly to African-American subgroup twice, Asian subgroup once, and people over the age of 60 subgroup twice. The first stage was trained on the last ten fully connected layers on the original data, and the second stage was trained with all layers set to trainable on the augmented data.

Experiment B:
Description: One-stage training with the same augmentations as Experiment A. The model was trained on the augmented data with only the last ten fully connected layers set to trainable.

Experiment C:
Description: Two-stage training with the same augmentations as Experiments A and B. The first stage was the pre-trained model downloaded from UAB, trained on only the last five fully connected layers on the original data, and the second stage was trained with all layers set to trainable on the augmented data.

Findings:

- Augmenting minority groups led to worse performance overall, increasing bias rather than reducing it.
- Over-augmentation may have led to overfitting on synthetic patterns, making the model less robust on real validation data.
- Removing 30% of the majority group’s samples (Caucasian) helped balance dataset distribution but did not significantly reduce bias in predictions.
- The one-stage training approach performed slightly better in terms of computational efficiency

TABLE 1: Comparing the results of the final model using data augmentation vs. the baseline results **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	NO	NO	0.1540	**0.1023**	**0.3441**	3.1021	0.9256	4.8119
b) Starting-kit	YES	NO	**0.0102**	0.3210	0.5005	**2.4936**	**0.8313**	**4.7332**
c) Extra 10 layers 2 stage	YES	NO	4.0059	2.4586	4.5119	33.493	11.1174	9.8547
d) Extra 10 layers 1 stage	YES	NO	4.007	2.4567	4.5014	33.4176	11.0957	7.76556
e) Extra 5 layers 2 stage	YES	NO	3.9873	2.4686	4.5033	33.4654	11.1061	15.84

5. FINAL REMARKS

Overall, we tried many different combinations of augmentations and training strategies, but in the end everything we tried worsened the starting-kit results by a lot. With more time and computational resources, we would try the ViT/Pytorch model to see if it would have performed differently. Additionally, we would test out more training strategies and different architectures, for example setting different random layers to trainable/not trainable for stage one training. The more we tried to augment the data, the worse results we got, which leads us to believe that less augmentations produce better results. But this also leaves us wondering, if we are augmenting so little, and the dataset with no augmentation works best, then what is the point of augmenting at all if it barely makes a difference in the distribution of the dataset anyways? We believe that the goal of the task, mitigating bias, would best be achieved by having a more balanced dataset in the first place, which is a very attainable goal as there are plenty of images of older people and people of color available that could have been included in the dataset, but were not. This is a recurring issue in machine learning models and obviously the main issue that we were trying to solve with this assignment, but data augmentation can only do so much when the baseline dataset is so skewed.