Intentions and aims

My initial intention was to build a classifier for identifying models of skin colour from real images of people facing the camera lens. The aim is to create a classifier which is extremely accurate and allow for several different skin colours to be identified.

Research into other related works

Research into similar work revealed several different Face recognition/ attribute classifiers. One work was created with the goal of Mitigating Bias and improving on existing Racial face datasets. They found that the numerous other datasets had a strong bias towards Caucasian faces, one study showed that some of the largest databases had a bias towards "lighter skin" faces (around 80%) compared to "darker" faces. They made the conclusion that this bias needed to be removed so they created their Face Attribute Dataset known as FairFace. This was developed by UCLA using Python, it performed better in almost every aspect than the other datasets. They achieved this by carefully curating their dataset with an "emphasis on balanced race composition". Meaning each of their 7 classes: White, Black, Indian, East Asian, Southeast Asian, Middle Eastern and Latino were well-balanced (**Figure 1.1**) and according to (Source) it is the largest of its kind. With this knowledge this project will attempt to maintain a well-balanced dataset to achieve better results.

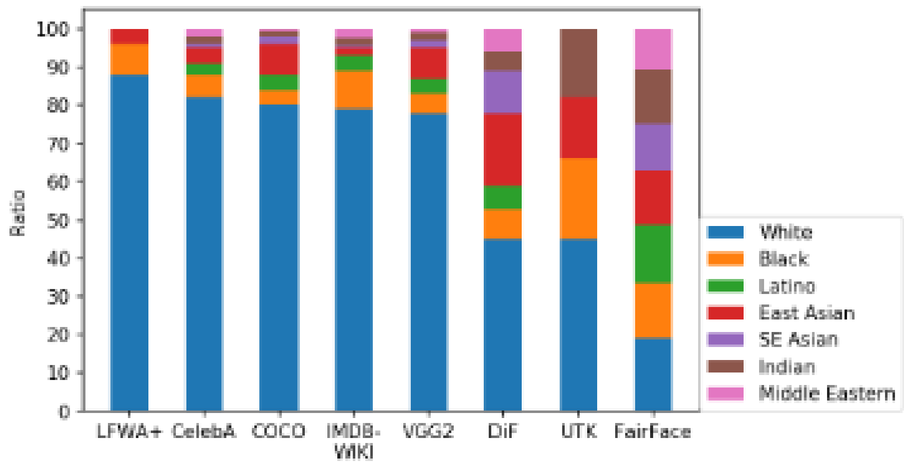

Figure **1.1**: Ratio compositions in the different datasets

As you can see (**Figure 1.1**) the FairFace dataset was not only well-balanced, but also has more classes than most of the other datasets. This was another step to remove the "white skin" bias and in turn *"performs better on novel data, not only on average, but also across racial groups, i.e more consistently*" they also explain that *"The inclusion of major racial groups, which have been missing in existing datasets, therefore significantly enlarges the applicability".* As this project wants to produce a similar application, it would be wise to include these different demographics to increase its overall applicability i.e., broaden its usability.

Another take away from this paper was the method in which to measure someone's skin colour. It is measured using Individual Typology Angle (ITA). This objective classifier is based on the colorimetric parameters CIELAB.

The L parameter is the luminance or the level of grey, from black (value 0) to white (level 100). The b* parameter is the yellow-blue component and the balance between yellow (positive value) and blue (negative value) that increases in relation to the intensity of pigmentation.*

Once the L* and B* have been determined the ITA can be ascertained in the following formula **Figure 1.2**. This allows for six different skin types to be classified:

- Very light **ITA° > 55°**
- Light **41° < ITA° < 55°**
- Intermediate **28° < ITA° < 41°**
- Tan **10° < ITA° < 28°**
- Brown **-30° < ITA° < 10°**
- Dark **ITA° < -30°**

Given that this application is attempting to remove as much bias as possible this formula will be used to objectively classify the projects dataset.

$$°\mathrm{ITA} = [\arctan(\mathrm{L}* - 50)/\mathrm{b}*]\ 9\ 180/3.14159.$$

Figure **1.2**: Individual Typology Angle formula

Curating the dataset

The dataset was initially collated by browsing online search engines like Google and Bing. The results were mixed, so a criterion was created, it follows:

- No watermarks or text
- Lighting must remain consistent between the images (Not bright or too dark)
- Their face should be facing the camera
- Nothing covering their face
- Avoid group photos

To automate this process a Google Chrome web extension called Image Downloader Continued was used. This allows users to search and filter out images e.g., dimensions, sorting order. Once the filters have been selected just scroll and select the images you want. As soon as the user has all the images just press download and it'll begin downloading all the images. This is an efficient way of collating a dataset, rather than manually downloading images individually. The dataset comprised of a total of 400 images, these were spilt into 4 classes, Asian, Black, Brown and White. Each class had 100 images, and these were split into 3 folders/sets training, validation and testing 70-15-15 respectfully. Whilst the dataset was complete there was an oversight, personal bias was not considered when assigning the images. No objective method was used to organize the different classes. Therefore, the dataset was useless and needed to be curated again.

Figure **1.3**: Imported image, red dot highlights where the LAB values are being obtained

Research was carried out to find a method (Refer back to research above) to objectively determine someone’s skin colour. Once the method (ITA) was discovered a python script was written in Google collab to speed up curating the new dataset. This script imported an image and plotted a dot onto the image Figure **1.3**. At the dots position the L* and B* values were collected. This data was used in the ITA formula to categorize the image into the corresponding folder, Figure **1.4** demonstrates how they are grouped. While this allowed for the dataset to be curated it was too laborious and was a suboptimal solution to create the new dataset. Due to the script being limited to reading one image at a time. Although there was consideration to adapt the script to sort through multiple images (folders) a far easier solution was uncovered. During the research phase a dataset with similar goals was retrieved, the FairFace dataset. This dataset split the different classes using a similar method which was used in the previously mentioned python script ITA.

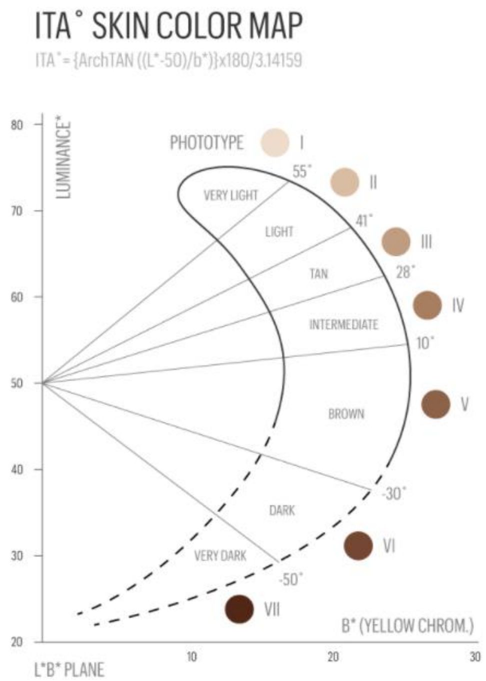

Figure **1.4**: Individual Typology Angle (ITA) map.

Since the FairFace dataset had already catergorised the data it was decided to use the same catergories to curate the dataset. Here is how the data was divided:

- Black
- East-Asian
- Indian
- Latino-Hispanic
- Middle Eastern
- Southeast-Asian
- White

Much like the initial dataset each class had 100 images, and these were split into 3 folders/sets training, validation and testing 70-15-15 respectfully.

Methods that were used

MLP

The model was initially trained using a Multi-Layer Perceptron network or MLP. This is a neural network where mapping between inputs and output is non-linear. They are comprised of; input layers, output layers and one or more hidden layers.

"Multilayer perceptron's are often applied to supervised learning problems: they train on a set of input-output pairs and learn to model the correlation (or dependencies) between those inputs and outputs." (Source)

This differentiates itself from the single perceptron network, which only solves linear problems.

Technical implementation

MLP

This classifier is written in Python, utilises the Pytorch library and is run on Google Collab.

The script firstly loads in the dataset by mounting to the google drive and then unzips the dataset dot zip. Next the transforms are defined this is, so the image data is in the correct format. It will resize, crop or even change the RGB values but in this case, they are normalized. This is applied to each folder training, validation and testing.

Results

Application of creative task

Conclusion

References

https://openaccess.thecvf.com/content/WACV2021/papers/Karkkainen_FairFace_Face_Attribute_Dataset_for_Balanced_Race_Gender_and_Age_WACV_2021_paper.pdf

https://wiki.pathmind.com/multilayer-perceptron

https://towardsdatascience.com/multilayer-perceptron-explained-with-a-real-life-example-and-python-code-sentiment-analysis-cb408ee93141