Artificial Intelligence: Deep Learning

Spring Semester 2021

Project Assignment

“Age Group Estimation Using CNN”

Cansu Moran

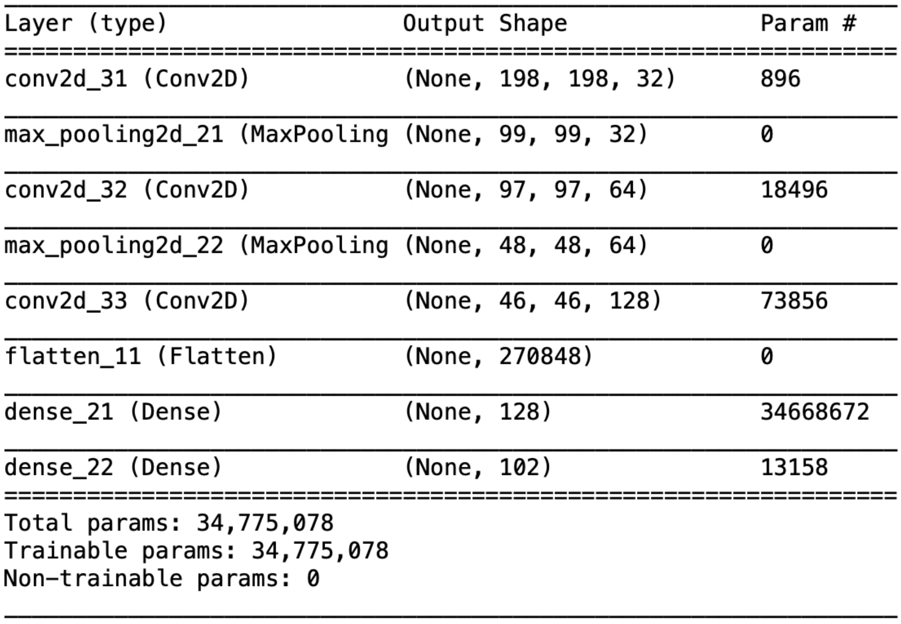
In this project, an age group estimation is done using Convolutional Neural Networks. As a dataset, UTKFace dataset[1] is chosen. In the UTKFace dataset, there are over 20,000 pictures with a single face in frame, with the size 200x200. The ages of the people in the pictures are given as the first number in the file name. In the dataset, there were pictures from age 1 to 116. However, it has been observed that after a certain age the number of samples decrease substantially. For example, when there are 177 pictures for age 15 there are only 5 images for age 93.

In all of the experiments, the same network structure is used. The only difference between the networks is the output channel number of the dense layer, as it depended on the number of categories in the dataset. The network used is inspired from the cat-dog classification network on Moodle[2]. Contrary to the cat-dog classification network mentioned, the network in this project uses three convolutional layers instead of four. In addition, since a multi-class classification with single label is being done, softmax activation function is chosen instead of sigmoid for the last dense layer. In addition, the output channel number of the last dense layer is modified according to the number of categories in the dataset.

The network includes 3 convolutional layers. The first layer has 32 filters, whereas the second one has 64 and the third one has 128. Since the category number is different for different experiments, the last dense layer has 102 output channels for the first experiment, 3 for second and 6 for last. In between the convolutional layers, max pooling is done to decrease sample size and make it easier to compute. The summary of the model used for the first experiment with 102 categories can be seen in Figure 1. Since we are making multi-class classification, categorical cross entropy is used for entropy measurements. To be able to use ImageDataGenerator, the data is separated under different directories where the label of the data in the folder is given as the folder name.

[1] http://aicip.eecs.utk.edu/wiki/UTKFace

[2] https://moodle.ruc.dk/mod/resource/view.php?id=322575



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
conv2d_31 (Conv2D)           (None, 198, 198, 32)      896       
_________________________________________________________________
max_pooling2d_21 (MaxPooling (None, 99, 99, 32)        0         
_________________________________________________________________
conv2d_32 (Conv2D)           (None, 97, 97, 64)        18496     
_________________________________________________________________
max_pooling2d_22 (MaxPooling (None, 48, 48, 64)        0         
_________________________________________________________________
conv2d_33 (Conv2D)           (None, 46, 46, 128)       73856     
_________________________________________________________________
flatten_11 (Flatten)         (None, 270848)            0         
_________________________________________________________________
dense_21 (Dense)             (None, 128)               34668672  
_________________________________________________________________
dense_22 (Dense)             (None, 102)               13158     
=================================================================
Total params: 34,775,078
Trainable params: 34,775,078
Non-trainable params: 0
_________________________________________________________________
```

Figure 1: Model summary of the CNN used in the first experiment with 102 categories

Initially, an exact age estimation is tried using the given network at Figure 1. There were 116 categories initially, since in the description of the dataset it was mentioned that the data had samples for ages from 1 to 116. However, after separating the data according to ages, it has been observed that there were no samples were certain ages such as 94. In addition, for some ages, there were only 1 image, which meant that we would have to use the same image both for training and validation. Since validating the model with the same data used for training might tamper with the accuracy of the model, all the ages with 1 or less samples were removed from the dataset. With this removal we ensured that for all the age categories given, there would be at least one sample for that age in training and validation data that are different. The initial dataset is split into training and validation data with a 70% to 30% rate respectively. The data is initially split randomly. However, in this case, we have realized that some ages had only samples in training data but not samples in validation data, especially the ages where there were only a few samples to begin with. Therefore, to ensure that both training and validation data has at least one sample for each age, one picture from each age is placed into training and validation data. For the remainder of the samples, the split is done randomly. In this experiment, the model wasn't able to train on the given data with high accuracy as seen on Figure 2 and Figure 3. For training data, the model was able to reach 20% accuracy whereas for validation data it was only able to reach 14% accuracy. These results are expected as the difference between for instance a 23 year old and a 24 year old is not something that can be trained. As a result, it has been observed that age estimation is not fit for a network that categorizes the given data, as age is more continuous than strictly categorical.

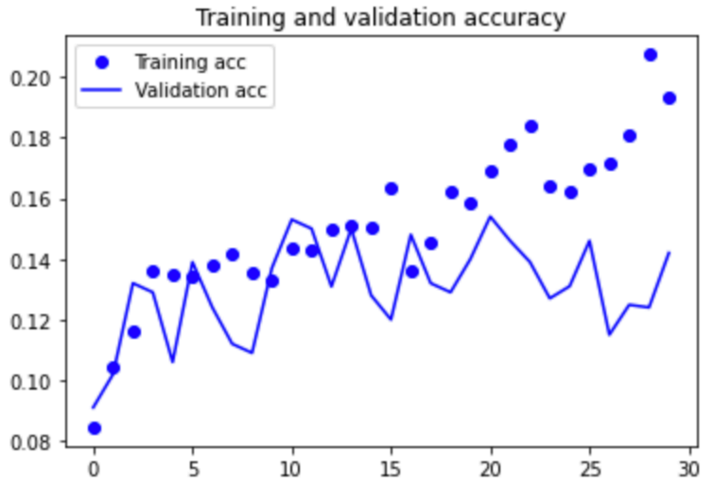


Figure 2: Training and validation accuracy for the first experiment

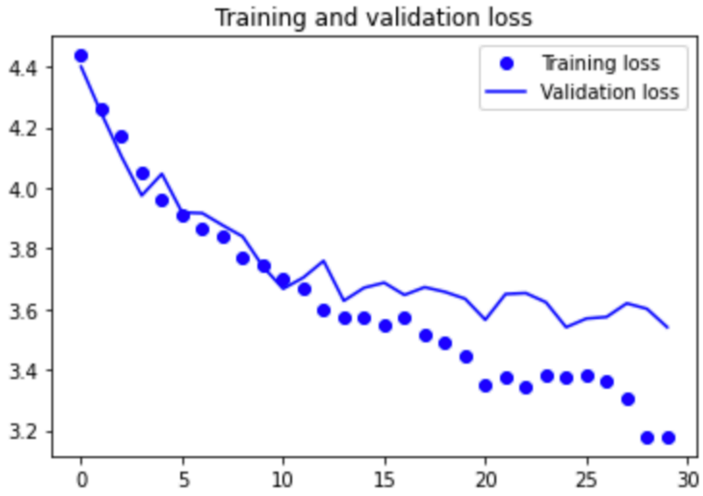
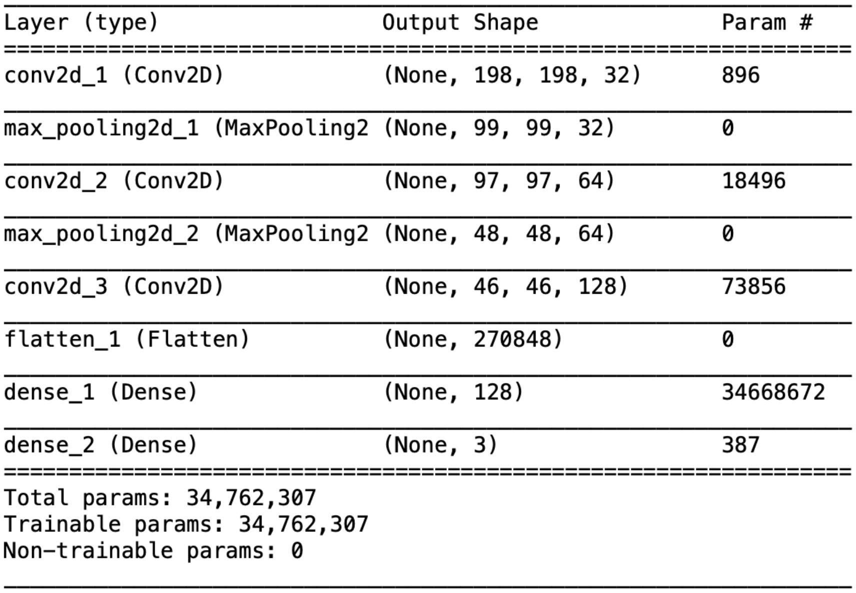


Figure 3: Training and validation loss for the first experiment

Since exact age estimation was not successful, a different approach has been experimented. Instead of training the model for all the different ages, it is trained for different age groups. The ages are separated into three groups: 0-14 ages are grouped as child, 15-64 ages are grouped as adult and 65+ are grouped as elder. The data is split with the same 70% to 30% rate for training and validation. Other than the last dense layer, all the layer structures are kept the same with the first experiment, as seen on Figure 4. On the last dense layer, the number of output channels is three since there are only three categories. Compared to the first experiment, the second network with three age groups performed extremely well. The accuracy and loss percentages for this experiment can be seen on Figure 5 and Figure 6. After training the network for these three categories, an accuracy above 90% was acquired both for training and validation.



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 198, 198, 32)      896
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 99, 99, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 97, 97, 64)        18496
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 48, 48, 64)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 46, 46, 128)       73856
_________________________________________________________________
flatten_1 (Flatten)          (None, 270848)            0
_________________________________________________________________
dense_1 (Dense)              (None, 128)               34668672
_________________________________________________________________
dense_2 (Dense)              (None, 3)                 387
=================================================================
Total params: 34,762,307
Trainable params: 34,762,307
Non-trainable params: 0
_________________________________________________________________
```

Figure 4: Model summary for the second experiment

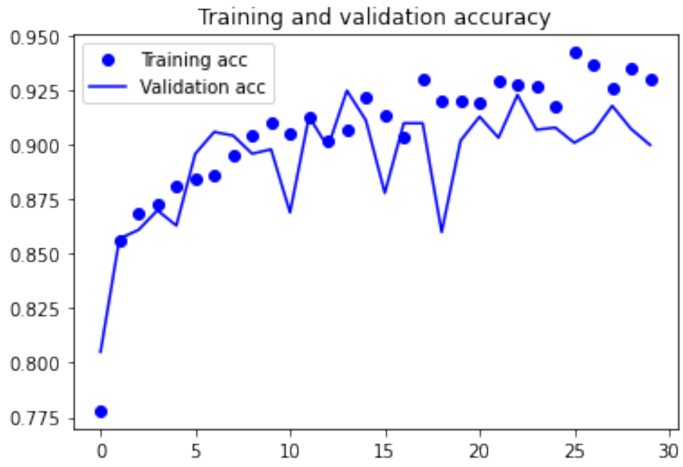


Figure 5: Training and validation accuracy for the second experiment

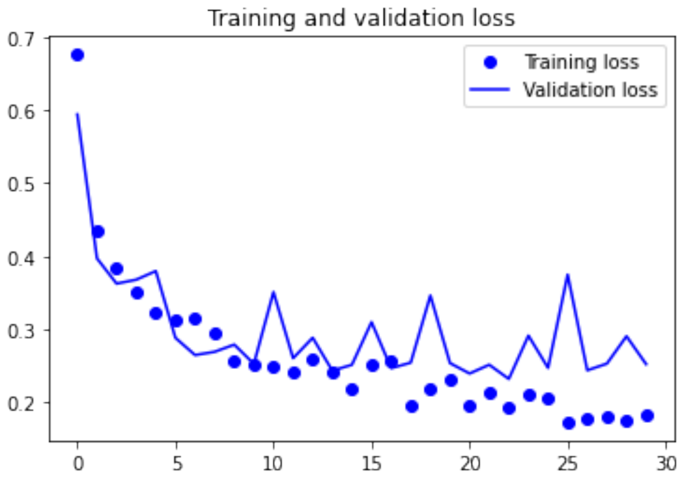
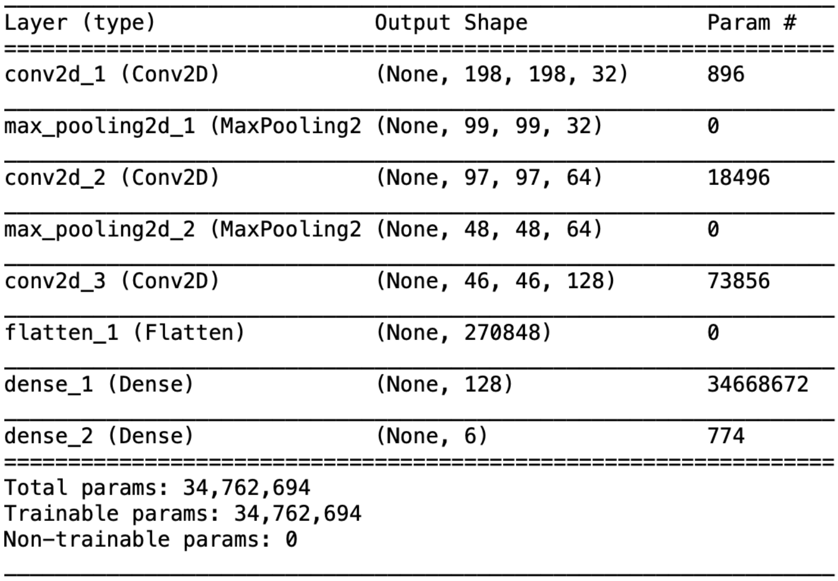


Figure 6: Training and validation loss for the second experiment

Since separating the ages into three groups was over-simplifying the classification compared to the first approach, a third experiment was done with an age group separation with less age gaps. For this experiment, the ages are split into six groups: 0-4 baby, 5-8 child, 9-17 teenager, 18-29 young adult, 30-59 adult and 60+ elder. The same network is used for this experiment as well. Only the output channel number of last dense layer is changed to 6 since there are 6 different age groups, as seen on model summary on Figure 7. The data is split the same 70% to 30% rate for training and validation. The results of the training and validation of this model can be seen in Figure 8 and Figure 9. After training the network for these six categories, an accuracy around 70% was acquired for training and validation. The accuracy of this experiment was expected to be in between the results acquired from the first and second experiment, as we have more categories compared to second experiment but less categories compared to exact age estimation done in first experiment.



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 198, 198, 32)      896
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 99, 99, 32)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 97, 97, 64)        18496
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 48, 48, 64)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 46, 46, 128)       73856
_________________________________________________________________
flatten_1 (Flatten)          (None, 270848)            0
_________________________________________________________________
dense_1 (Dense)              (None, 128)               34668672
_________________________________________________________________
dense_2 (Dense)              (None, 6)                 774
=================================================================
Total params: 34,762,694
Trainable params: 34,762,694
Non-trainable params: 0
_________________________________________________________________
```

Figure 7: Model summary for the third experiment

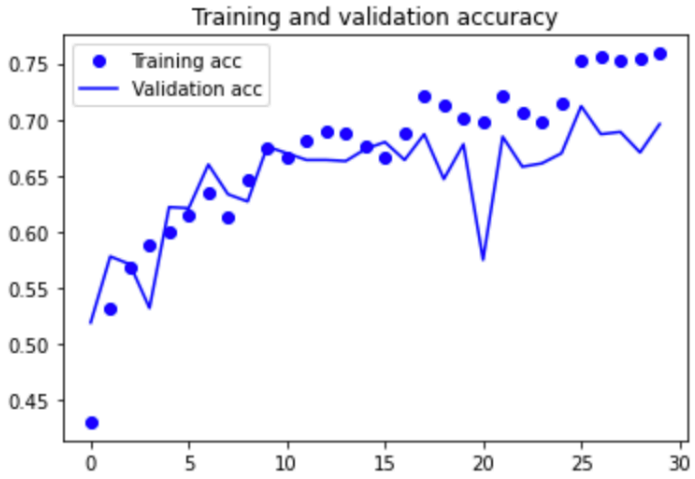


Figure 8: Training and validation accuracy for the third experiment

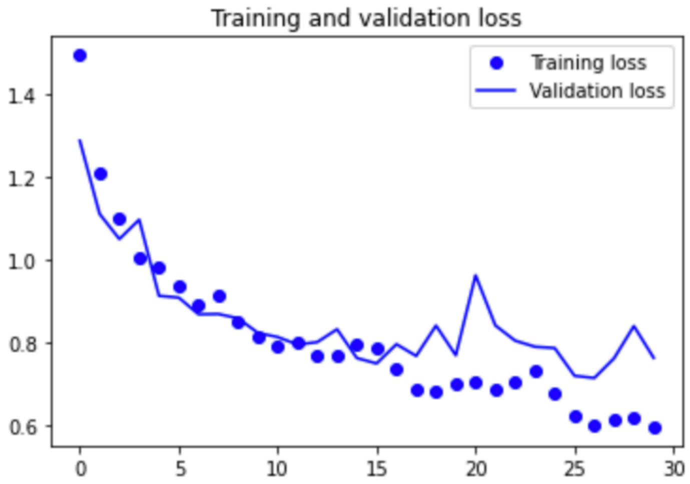


Figure 9: Training and validation loss for the third experiment

As a result, we have seen that categorical approach is not fit for exact age estimation, as the difference between close ages might not be something that can be noticeable, even for a human eye. Since age-face relationship is more continuous than categorical, the model was unable predict the ages accurately. However, as seen from the results, the network was able to perform well when the ages were separated into groups.

As mentioned before, in the dataset, there was a difference between the number of samples for younger versus older ages. Therefore, we can assume that the model can give better predictions for younger age groups compared to older age groups. In order to improve the model to predict all age groups with a similar accuracy, the sample sizes can be equalized by adding more elderly pictures.

In its current form, the network performs quite accurately. However, the layer structures can be changed to see how the accuracy changes. For example, certain parameters such as the number of filters can be changed and the number of convolutional layers in the network can be increased.

Appendix

UTKFace dataset: http://aicip.eecs.utk.edu/wiki/UTKFace

Codes: https://drive.google.com/drive/folders/1LTNtbpIGAdIvTcmwuheNf2UsutyswweI?usp=sharing