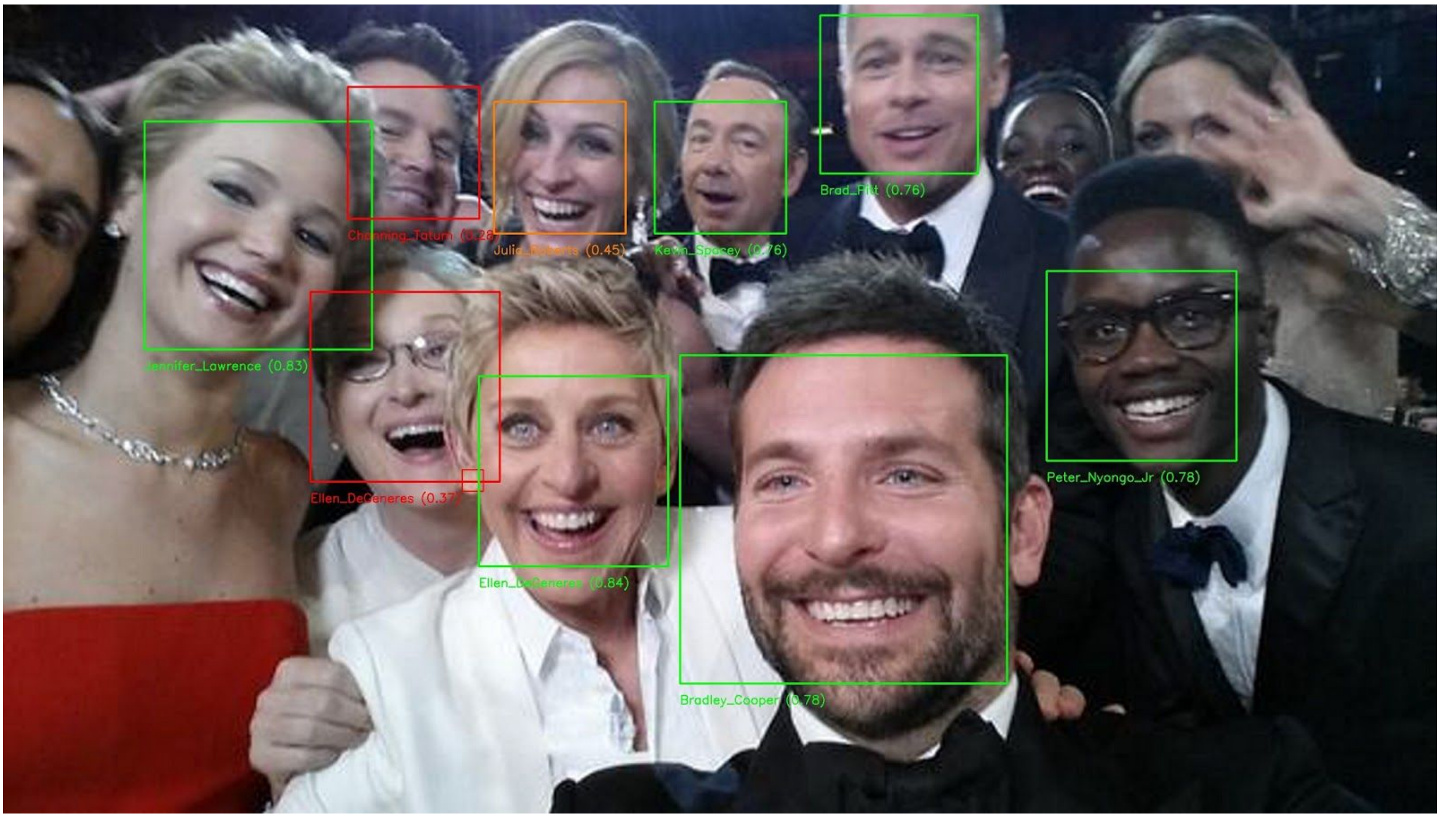
SDS M3 2019 - Group Assignment

Deep Learning & Artificial Intelligence for Analytics

Subject: Gender recognition



Aalborg University - AAU

Cand.IT - IT-Management - 9th semester

Group members:

- Kristian Stavad
- Rasmus Simmelsgaard Hye
- Lasse Hede

AALBORG UNIVERSITY
DENMARK

Introduction:

When scouting for a subject for this final deep learning assignment, different subjects and types of analysis was considered. We knew that we wanted to work with image recognition in some sort of way, because we find the technology rather interesting and useful for a vast amount of use cases. Image recognition uses artificial intelligence technology to automatically identify objects, people, places and actions in images. Image recognition is used in different business or personal objectives, and can help solve problems regarding identity authentication, healthcare or surveillance and security matters, to name a few. Through the lectures of the course, different image recognition techniques and types of codes has been presented, giving us the needed tools for an analytic task like we will present in this rapport where the objective is to classify the gender of people by letting the neural network look and learn from a dataset of images and attributes. We hope that you as the reader will enjoy our work. And now, let the fun begin!

Data Collected:

This dataset consist of more than 200.000 face images of various celebrities, with more than 10.000 unique identities and 40 attributes per image, names and genders is not included. This dataset is great for training and testing models for face detection, particularly for recognising facial attributes such as finding people with brown hair, are smiling, or wearing glasses. Images cover large pose variations, background clutter, diverse people, supported by a large quantity of images and rich annotations. The dataset was downloaded from kaggle.com.

Google Colab	https://colab.research.google.com/drive/1csLtkv64jyxBAOcZ3LOPN3_EUwh1eJv2#scrollTo=0xCGcsmZ148D
Github repository	https://github.com/LassehedeSDS/SDS-M3-2019---Group-Assignment-
CelebA Dataset	https://www.kaggle.com/jessicali9530/celeba-dataset
Getting access to Colab notebook and the dataset:	We experienced some problems mounting the dataset to Colab, after a period of inactivity the data would get deleted. To solve this we uploaded the dataset to a public google drive and from here imported it into Colab. To access the dataset we recommend that you download the dataset from the link below and

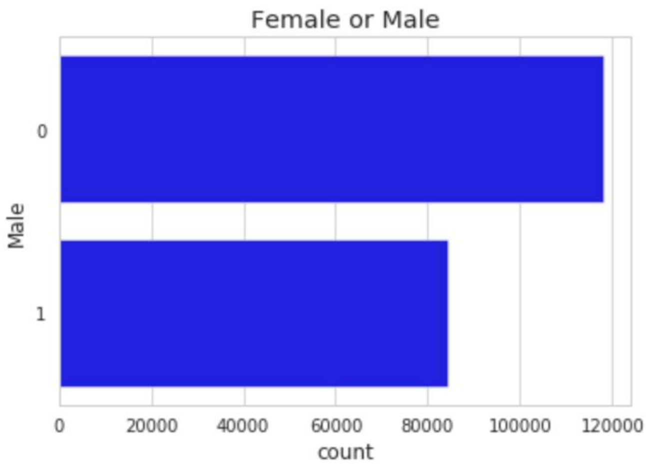
	upload it to your own google drive, and import it from your own google drive into colab. As soon as the data is uploaded to your own root folder in Google Drive, the code should do the rest and work as intended. https://drive.google.com/drive/folders/13dP-GafbdySROcLUS7qAaXukrPi0JcG-?fbclid=IwAR3BcBrabtCkvUsAwMQAW09c4LaX3-t-iVTUviybvnPt3cZtNK_WAhCX4AE

Data exploration:

In the first step, we explore our dataset, and make changes to the data, in order to prepare the data for further analysis. This includes importing the necessary libraries, changing dataset values and general exploration of the datasets attributes.

We wanted to see an example of one of the pictures in the dataset. To do this we printed out a random picture *'000409.jpg'* from the list. We also wanted to try and print some attributes to the given picture, to see if the attributes matched with the person on the picture. In this case we checked the attributes *'Bald'*, *'Male'* and *'Young'*. Below is the example picture with the given attributes, by doing a visual check we can see that it is in fact a younger male on a picture, who is not bald. The example printout was a success and the attributes matched the picture.

The overall goal of this analysis is to build a model that can recognize and classify the gender of a given person, by looking at pictures of people in different settings. In this regards it was interesting to visualize the total amount of males and females portrayed of the pictures. To get an insight of this distribution, we printed out a barchart with boolean values *'0'* and *'1'*. A *'1'* value equals to 'Male' and a *'0'* value equals to 'Female'. The result showed that there are ~118.000 pictures of males and ~84.000 pictures of females in the dataset.



Splitting the dataset into training, validation and test:

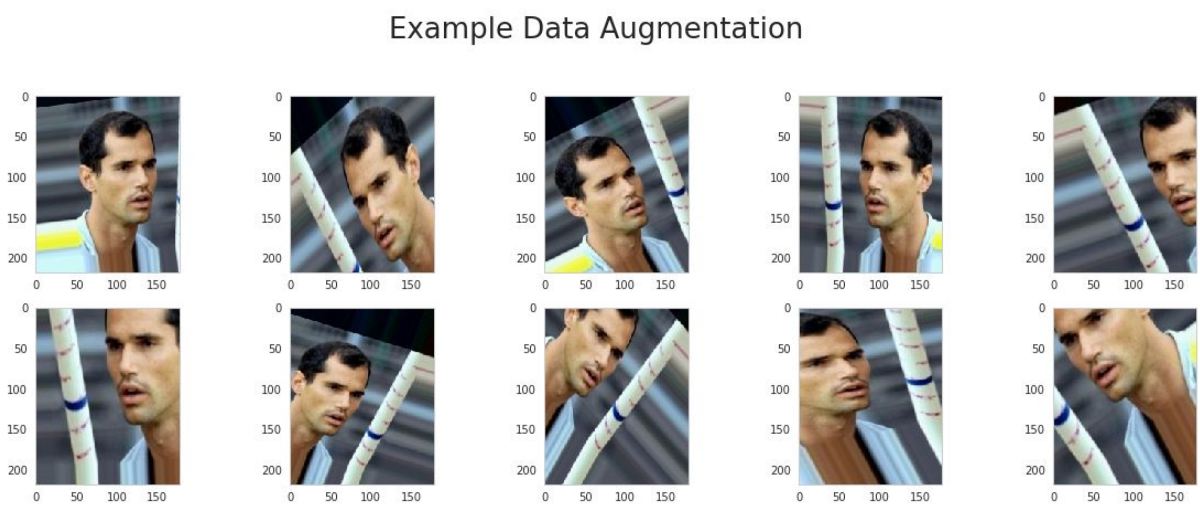
For this step we divide our dataset into three subsets of the dataframe, which is training, validation and test samples. In order to reduce time execution, we limit our splits to:

- **Training:** 20.000 images
- **Validation:** 5.000 images
- **Test:** 5.000 images

This way we train our data on ~66% of the dataset, validate on ~17% of the dataset and test on ~17% of the dataset. The splitting of the data is predefined in the 'list_eval_partition.csv' file.

Data augmentation:

In order to be able to make better predictions, we process the images with modifications to the original ones, by mirroring, stretching, scaling the images. The model will this way learn from these different variations, resulting in better predictions on images it has never seen before. Below we showcase the example picture, and how this image will look like after data

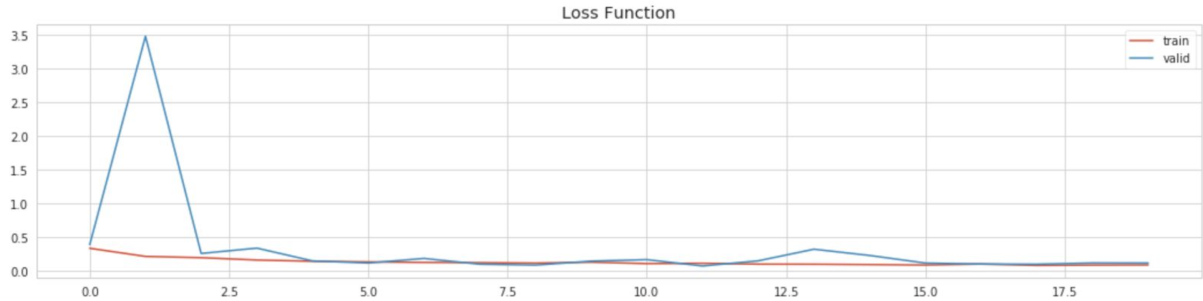


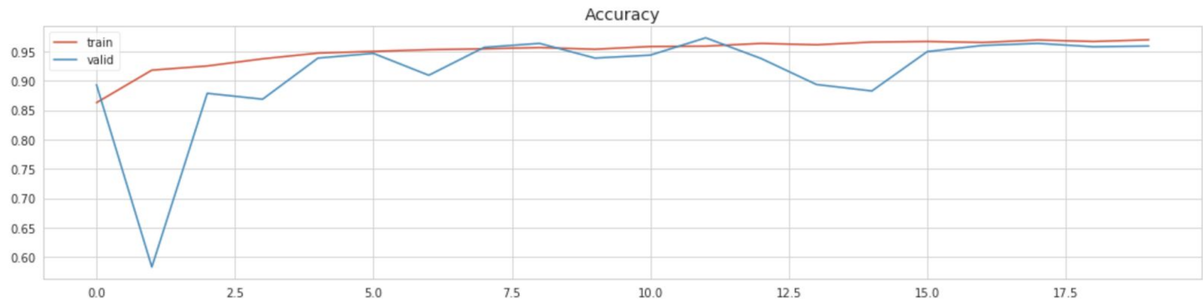
augmentation, we plotted 10 augmented images to observe the behaviour and result.

Building the model, Gender Recognition:

We decided to use the transfer learning model approach to solve our problem. This approach allows us to retrain the final layer of an existing model We chose the InceptionV3 model developed by Google and trained on more than a million images from Image.Net. This decreases the time we had to use on training the model, and we therefore simply had to tune the hyperparameters to match our problem and feed it our image data. The model uses different layers on the same time, to produce more precise data.

Executing this model on our image data, produces a ‘loss function’ and ‘accuracy’ graph, which shows, that the model has comparable performance on both the training and validation datasets. If the two lines start to depart from each other, this is a sign to stop training at an earlier epoch. The model shows, that through learning, we could stop somehow at a number of 15 epochs, since at this point the graphs seem to flatten out.





The final model evaluation shows, that we get a ‘Test accuracy’ of 96,2% and a ‘f1_score’ on 96,15%. The f1_score is a measure of the test accuracy.

```
Model Evaluation
Test accuracy: 96.2000%
f1_score: 0.9615384615384616
```

Conclusion:

The model successfully recognize the gender giving certain picture with **96% of accuracy**. Nevertheless, there are some limitations detected and opportunities for further improvements of the model:

- Improvement: We could train the algorithms with the entire dataset of images. Due to computational limitation, the model was train on a subset of images. Having the appropriate computational resources, the model can be trained on all the images. This will make the algorithm to learn from different context of the picture giving it more experience in order to predict better on new, never before seen images.
- Use difference structures for the CNNs. This approach could give better performance to the model, is an expensive task anyway, as the model can be measure on the test data set after is trained, and this takes time and computational resources.
- Looking at the pictures from the CelebA Dataset, most of the pictures are a close-up to the face of the subject, this leads the model to learn from this type of pictures, and in situation where the subjects is a smaller portion of a picture, the model could not perform well. To deal with this, more sophisticated preprocessing data can be added or complement the data set with pictures that are not entirely based in close-up to the face of the subject.