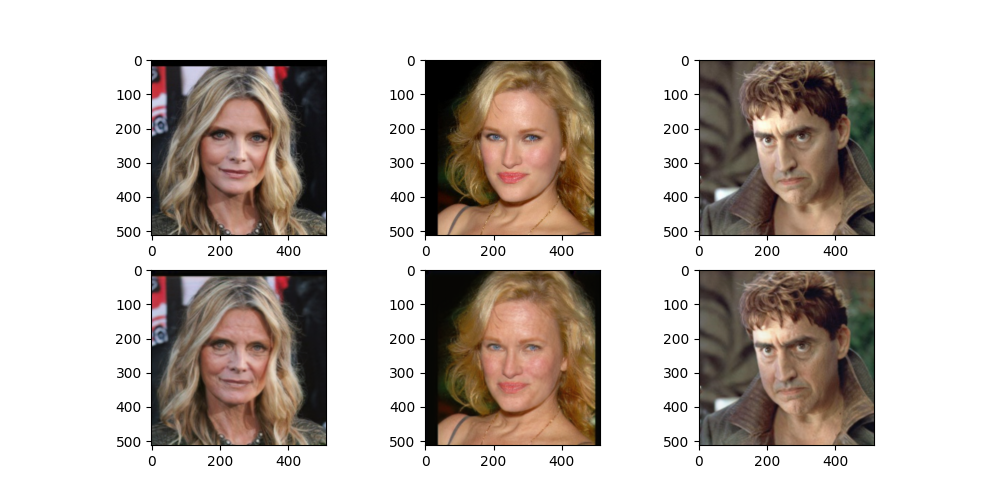
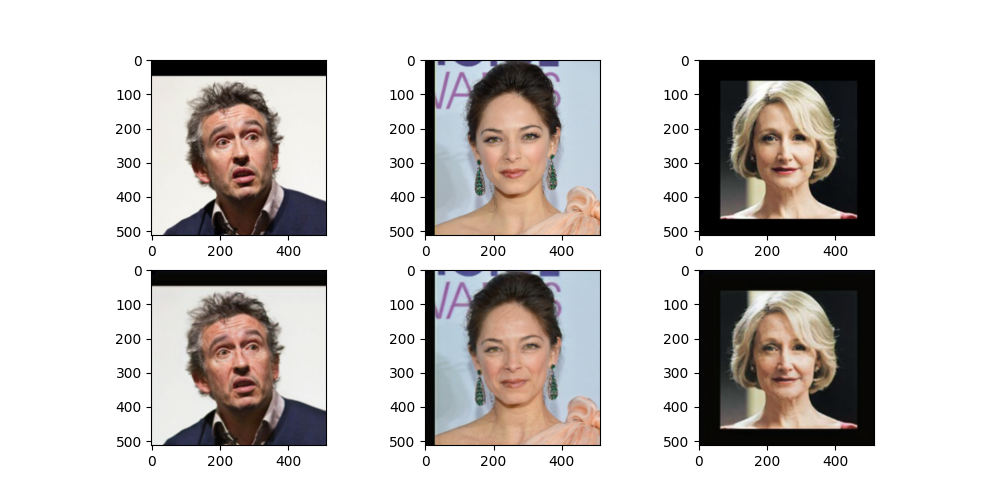
**Fast-AgingGAN**

This repository holds code for a face aging deep learning model. It is based on the CycleGAN, where we translate young faces to old and vice versa.

**Samples**

Top row is input image, bottom row is aged output from the GAN. [](https://user-images.githubusercontent.com/4294680/86517626-b4d54100-be2a-11ea-8cf1-7e4e088f96a3.png) [](https://user-images.githubusercontent.com/4294680/86517663-f5cd5580-be2a-11ea-9e39-51ddf8be2084.png)

**Timing**

The model executes at 66fps on a GTX1080 with an image size of 512x512. Because of the way it is trained, a face detection pipeline is not needed. As long as the image of spatial dims 512x512 contains a face of size 256x256, this will work fine.

**Demo**

To try out the pretrained model on your images, use the following command:

python infer.py --image\_dir 'path/to/your/image/directory'

**Training**

To train your own model on CACD or UTK faces datasets, you can use the provided preprocessing scripts in the preprocessing directory to prepare the dataset. If you are going to use CACD, use the following command:

python preprocessing/preprocess\_cacd.py --image\_dir '/path/to/cacd/images' --metadata '/path/to/the/cacd/metadata/file' --output\_dir 'path/to/save/processed/data'

If using UTK faces, use the following:

python preprocessing/preprocess\_utk.py --data\_dir '/path/to/cacd/images' --output\_dir 'path/to/save/processed/data'

Once the dataset is processed, you should go into  configs/aging\_gan.yaml and modify the paths to point to the processed dataset you just created. Change any other hyperparameters if you wish, then run training with:

python main.py