AGE CLASSIFICATION using CNN

Project by Nemanja Vujadinovic

PROBLEM

The idea of this project is to train neural network for people **age recognition**. Neural network output for one person image should be young or old. It's not easy to determine if someone is young or old, so the neural network will also output the probability of person being young.

USED DATA

Dataset used for building neural network is ***CelebA***. It contains over 200000 portrait images of celebrities with attribute labels. Our main attribute is ***young*** being a key feature for training and testing neural network.

Example of one image from dataset

NEURAL NETWORK

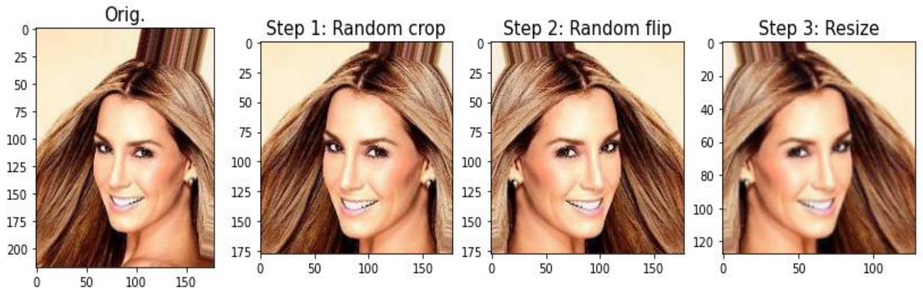
The chosen neural network is convolutional - **CNN**.

```
Sequential(
  (conv1): Conv2d(3, 32, kernel_size=(5, 5), stride=(1, 1), padding=(1, 1))
  (batchnorm1): BatchNorm2d(32, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (relu1): ReLU()
  (pool1): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (dropout1): Dropout(p=0.6, inplace=False)
  (conv2): Conv2d(32, 64, kernel_size=(5, 5), stride=(1, 1), padding=(1, 1))
  (batchnorm2): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (relu2): ReLU()
  (pool2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (dropout2): Dropout(p=0.4, inplace=False)
  (conv3): Conv2d(64, 128, kernel_size=(5, 5), stride=(1, 1), padding=(1, 1))
  (batchnorm3): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (relu3): ReLU()
  (pool3): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (conv4): Conv2d(128, 256, kernel_size=(5, 5), stride=(1, 1), padding=(1, 1))
  (batchnorm4): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  (relu4): ReLU()
  (pool4): AvgPool2d(kernel_size=4, stride=4, padding=0)
  (flatten): Flatten(start_dim=1, end_dim=-1)
  (fc): Linear(in_features=256, out_features=1, bias=True)
  (sigmoid): Sigmoid()
)
```

Model of implemented CNN with layers shown

DATA AUGMENTATION

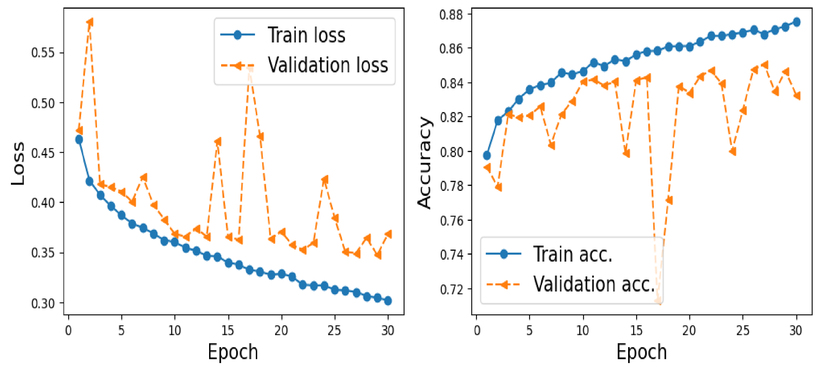
For better results on smaller training dataset, we are using **data augmentation**.



Random transformations on one of the images

RESULTS

There have been **26000** images for training, **7200** for validation and **3600** for testing datasets.

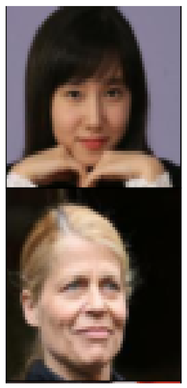


Loss and accuracy graphics for training and validation through epochs

The testing accuracy is **0.8512**.

Ground truth: Young
CNN classification: Young
with the percentage of youngness 98%

Ground truth: Old
CNN classification: Old
with the percentage of youngness 2%



Visual representation of the results

CONCLUSION

It is important to acknowledge that over 70% of dataset are image with attribute young labeled as true, which will make getting accuracy above 85% harder. Also, it is difficult for a human to judge if the one is young or not, let alone for a computer. Still, all of this can be overcame using greater training dataset, as well as trying to minimize overfitting, which will contribute to better results, yet our neural network produced good results considering hardware and resources availability.