

Report Title Here

Radhika Mattoo
rm3485@nyu.edu

Mohammad Afshar
ma2510@nyu.edu

December 21, 2016

1 Introduction

Our objective in this project was to achieve a relatively low error in face recognition by exploring and training variations of existing successful architectures. Our models were influenced specifically by Facebook and Google's DeepFace and FaceNet architectures, respectively, as well as the winning architecture from the 2011 German Traffic Sign Recognition Benchmark. We used a subset of Stefan Winkler's FaceScrub dataset to give us 41.6 thousand training images and tested our models using Labeled Faces in the Wild. Despite resource and time constraints, we created, trained, and tested these different architectures in order to compare and analyze the resulting rates in face recognition.

2 Problem Summary

The problem we aimed to solve using our models was facial recognition with 'same-not-same' classification. Thus, we trained a convolutional neural network on our dataset, and used Cosine similarity during testing to determine if two images passed through our trained network were of the same face.

3 Baseline Method

Mohammad

4 Architectures

Mohammad

4.1 Model 1

4.2 Model 2

4.3 Model 3

5 Obstacles

5.1 Data

Our largest obstacle with respect to data was simply finding enough viable images to use for training. While Facebook’s DeepFace model was trained on a private dataset of 4.4 million images, for our training dataset we were left to our own devices. After spending much time searching online for a large dataset of colored face images, we discovered Stefan Winkler’s FaceScrub dataset of 100 thousand celebrity images, each including labels and a download URL. While this dataset was the best and largest we came across, downloading these images, then aligning and cropping them proved to be too time-consuming and inefficient. To preserve time for actually training our different models, we decided to download 41.6 thousand images in total from FaceScrub. Thus, already hindered on one of the most important aspects of the project, we knew our results would be considerably subpar to those of DeepFace, FaceNet, and the Recognition Benchmark.

5.2 Resources and Time

In addition to lack of data, we encountered issues with our resources and time. Initially, our plan was to use NYU’s High Performance Computing (HPC) clusters for our project, thus giving us the opportunity to implement, train, and test many different models. Unfortunately, over the past month, HPC has been unreliable in its service; it would either reject or queue our jobs for long periods of time. Thus, we were forced to use our own, significantly weaker, laptops to collect and clean our data, and train and test our models. Additionally, attempting to install qlua on different computers, such as our more powerful desktops, lead to the discovery that qlua’s build was broken at some point over the past month. Finally, we planned on imitating Facebook’s DeepFace architecture exactly, and train it with different parameters to see its effects on our results. Torch, however, does not offer locally connected layers, which an integral and essential component to the architecture’s success. With more time to collect proper data and explore different libraries, such as TensorFlow for more diversity, and access to more

reliable and available computing resources, we would have the opportunity to explore more models in much more depth.

6 Results

Mohammad

7 Conclusion

-Deadass repeat the intro

8 Future Works

Mohammad

9 References