# MACHINE LEARNING ENGINEER NANODEGREE

## Capstone Proposal

### Face Recognition: *Look* How Far We've Come

John Quinn
March 6th, 2018

### Domain Background

Computer Vision ("CV") today is a substantial subset of Artificial Intelligence ("AI".) Other areas of AI, including robotics and its application to autonomous vehicles, moreover, are inextricably related to CV; an autonomous car that can't "see" traffic cannot, of course, drive in traffic.

The seminal CV paper, at least with substantial scientific rigor, was delivered at MIT (Roberts, 1963). In the half-decade since, progress has been formidable, in particular since Google's publishing/making open source TensorFlow[1] ("TF") just over two years ago. Applications of TF continue to astound (Isola, Zhu, Zhou, & Efros, 2017).

At present I have completed all of my required classes and am working on my thesis towards my M.S. in Software Engineering, concentrating on math, statistics, and AI, at Harvard. My pending thesis is a proposed application of CV, e-commerce via customer picture upload – thus my interest in CV, both from an academic, historical perspective and from that of a prospective data scientist.

### Problem Statement

The somewhat classic Eigenfaces[2] problem with its related database has become a rite of passage for a CV student. The dataset includes 1,288 samples, JPEG files, each encoding 1,850 features A principal component analysis ("PCA") transformation is performed, reducing the feature space by approximately an order of magnitude, and a support vector machine ("SVM")/classifier is employed for the top 150 of the 966 (75%) training set images.

I hope to show that more recent approaches can outperform this PCA-SVM approach. Specifically, I plan to compare the PCA-SVM results to those resulting from employing Haar cascades (Viola & Jones, 2004). I then hope to try one or more deep learning ("DL") approaches, such as using a convolutional neural network ("CNN".) In each case, I may test other means, such as data augmentation, to better results.

[1] http://www.tensorflow.org
[2] http://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_lfw_people.html#sklearn.datasets.fetch_lfw_people

The Eigenfaces problem's PCA-SVM results provide several metrics. I hope for my thesis/putative business idea to be used in e-commerce, most commonly on returning customers. As such, I believe precision (i.e., true positives, divided by the sum of true positives and false positives) is the most relevant metric.

In identifying a *product*, recall (i.e., true positives, divided by the sum of true positives and false negatives), would likely be appropriate, but in identifying a *customer* (via his or her face – the case at hand) it is more important to incorporate false positive than false negative score diminution. Intuitively, a customer will be less put-off not being recognized than being falsely identified by an algorithm.

## Datasets and Inputs

The dataset will be the Labeled Faces in the Wild ("LFW") dataset employed by the Eigenfaces problem.[3] Its 1,288 JPEG photos of original dimensionality 1,850 should suffice.

It is possible, however, that additional data may be required. In the event that this is so and that I am unable to generate synthetic samples, such as with a deep convolutional generative adversarial network, I will restrict any such additional data to royalty-free, publicly-available sources. In that case I will also provide within the Jupyter notebook, the centerpiece of my ultimate deliverables, code that can be executed directly there to obtain such data.

## Solution Statement

I plan to deliver precision greater than the 84% achieved via the original PCA-SVM approach. Haar cascades work well on reasonably clear, forward-facing images of faces only (i.e., not set within a scene), and the LFW dataset seems mostly comprised of the same. Data augmentation and other skills learned in this nano-degree should also provide means to improve that score if appropriate.

The more recent DL approaches, moreover, should perform better still, albeit with the proviso that paucity of data/samples may need to be rectified. The bespoke conditional adversarial network from the Berkeley AI Research Laboratory, e.g., consistently placed in the top 1% in large dataset image classification contests. (Isola, Zhu, Zhou, & Efros, 2017).

## Benchmark Model

The benchmark model for this project is the Eigenfaces PCA-SVM one. Thus, success for this project will be defined as achieving precision of greater than 84% on the LFW dataset.

[3] http://vis-www.cs.umass.edu/lfw/lfw-funneled.tgz

## Evaluation Metric

Once again, achieving greater than 84% precision on the LFW dataset will be the evaluation metric for this project.

## Project Design

For easy presentation, all primary code will be included in a summary Jupyter notebook. Helper classes and/or functions not central to the project, if any, will be incorporated in one or more other Python files.

I intend to start by employing code to obtain the LFW dataset and make it readily usable within the notebook. For comparative purposes and to allow the user to run his or her own what-if calculations and/or visualizations I will also incorporate the PCA-SVM solution, albeit converted to Python3.

Then I intend to use openCV's Haar Cascade face identifier.[4] Should the results of this approach not suffice I may try other methods, such as data augmentation.

Finally, I will try DL as a solution to Eigenfaces. I plan to start with transfer learning and a somewhat straightforward CNN and then see if I can improve results with more sophisticated approaches.

## Bibliography


Isola, P., Zhu, J.-Y., Zhou, T., & Efros, A. A. (2017, November 22). *Image-to-Image Translation with Conditional Adversarial Networks.* Retrieved from Cornel University Library: arXiv.org

Roberts, L. (1963). *Machine Perception of Three-Dimensional Solids.* Massachussets Institute of Technology, Electrical Engineering. Cambridge, Massachussetts: ResearchGate.

Viola, P., & Jones, M. (2004, May). *Rapid Object Detection Using a Boosted Cascade of Simple Features.* Retrieved from Mitsubishi Electric Research Laboraties: http://www.merl.com/publications/docs/TR2004-043.pdf


[4] https://docs.opencv.org/3.3.0/d7/d8b/tutorial_py_face_detection.html