Problem Statement

The question is this: given a twitter profile picture, can we build a model that determines whether they are wearing eyeglasses in the profile picture? My client would be any sort of eyeglass company (e.g., Warby Parker), that wants a more automated way of targeting potential customers. The company would search through different Twitter users and, if the model determines they wear eyeglasses, the eyeglass company would target these users with advertising or speciality deals. The company would thus be able to more effectively target potential customers (those that wear glasses) and improve sales. Additionally, the modeling dataset contains a variety of other attributes, e.g., wearing a hat, having a moustache, having bangs, etc., with which we could use to retrain the CNN and target different users on Twitter. A company that sells hats or facial care products for men would thus also be able to use this model to focus their advertising efforts on Twitter users that wear hats, have moustaches/beards, etc.

Data Cleaning and Wrangling

The primary dataset for this project is the Celebrity Faces Attribute (CelebA) dataset, which “is a large-scale face attributes data with more than 200K celebrity images, each with 40 attribute annotations.” Much of the preprocessing and data wrangling steps for this dataset were done following the code in this GitHub repo, where a convolutional neural network (CNN) was trained on the same dataset (CelebA) to determine whether the celebrity in the image is wearing eyeglasses or not. The images in the CelebA dataset are of size 178 x 218 and are ≤10 KB each. The first step in this process is gathering all the images in the dataset where someone is wearing eyeglasses--a total of 13,193 images from the total 202,599 image dataset, or ~6.5%. We would like

our total, modeling dataset to consist of 80,000 images, so we gather the remaining 66,807 non-eyeglasses images. After gathering all the images, we shuffle their order and use a custom face aligner to realign the celebrity images such that the alignment process between the CelebA dataset and the later Twitter-profile-picture dataset is consistent. In some instances, the face-aligner fails to capture any faces in the celebrity image and thus fails to align them--these instances are skipped over and not included in the modeling dataset (removes ~3.85% of the 80,000 images, leaving us with 76,914 images for modeling). The images are then converted to grayscale and resized to 28 x 28. The images along with their corresponding labels (1 for eyeglasses, 0 for no eyeglasses) and image file names are saved together in a numpy file.

To construct the Twitter-profile-picture dataset, the avatars.io service along with Python's `requests` module was used to first obtain the profile pictures from Twitter. I gathered 17 Twitter profile pictures and hand labeled them 1 if they wear glasses and 0 if they do not. These images were then put through the same pipeline as the CelebA images--first align the face and resize to the same size as the images in the CelebA dataset (178 x 218), then convert to grayscale, and finally resize to 28 x 28. This process is shown for a few examples in **Figure 1**.

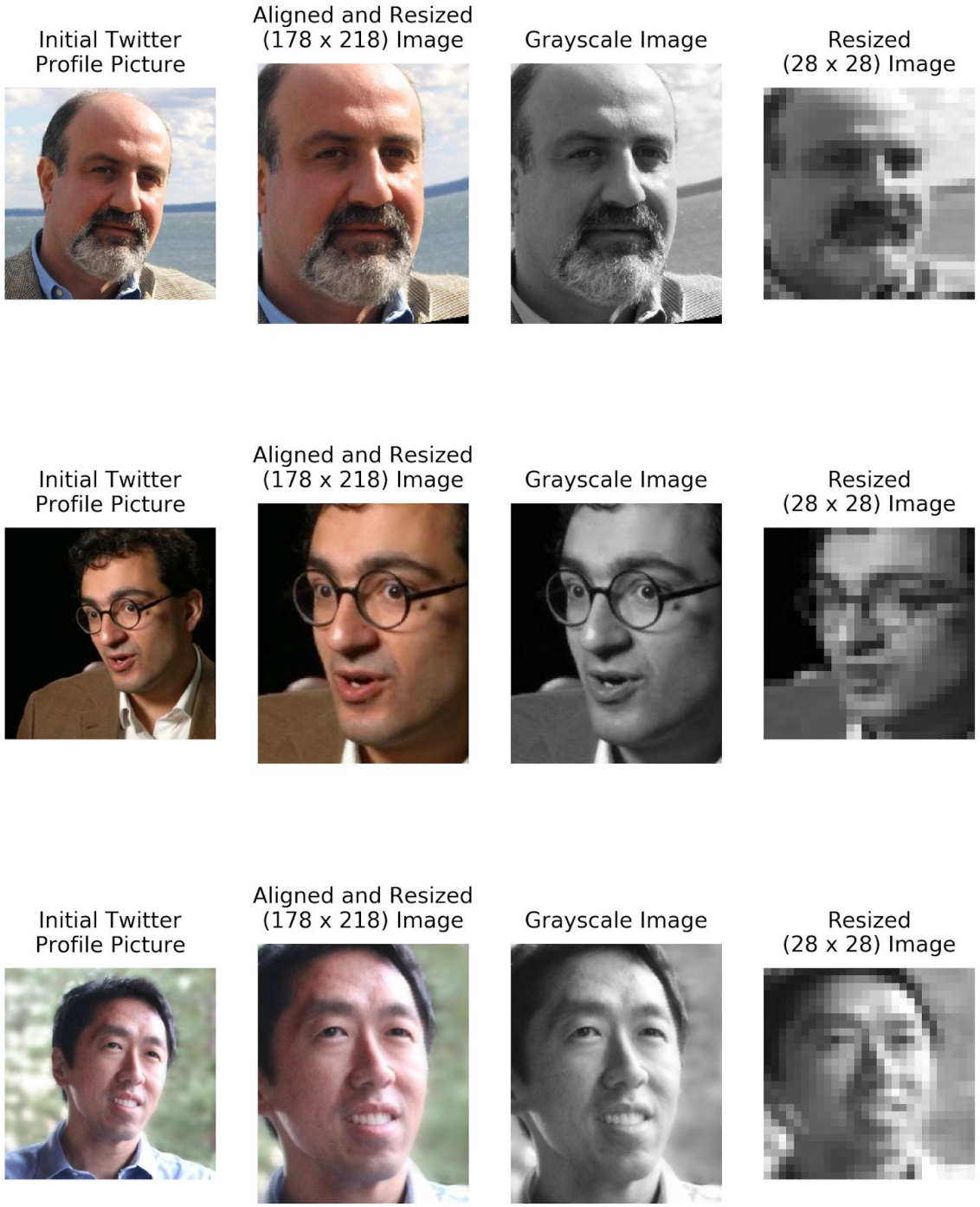

Figure 1. Three example of the image preprocessing for twitter profile pictures. From left to right: first we have the initial Twitter profile picture, then the alignment and resizing to the CelebA

dataset, then converted to a grayscale image, and finally resized to 28 x 28, which will be used for modeling. From top to bottom, the Twitter users are @nntaleb, @SimonDeDeo, and @AndrewYNg.

The dataset is now ready for modeling.