Finding Your Celebrity Look Alike

Chase, Davis, Amanda Jacquez

chased@stanford.edu, ajacquez@stanford.edu

December 18, 2019

Abstract

As humans, we have a unique fascination with faces. Faces are the part of the human body most often exposed, and the primary method used for recognition. Because of this, our faces are often considered significant parts of our identity. Today's society places a large emphasis on celebrities. People often compare and contrast themselves with the celebrities that they see on television and hear on the radio. Because of this high emphasis on celebrities, we wanted to implement a project around the topic of facial recognition that would help users find their celebrity look-alike. Facial recognition is an important tool that is ubiquitous in modern society. It is used everywhere, from FaceID on modern iPhones to auto-tagging on social media, airports, shopping centers and even law enforcement. Inspired by the fact that celebrities and facial regonition tools maintain an emphasized and increasingly important presence in society, we were motivated to create a project that address these individual components.

The goal of this project is to employ computer vision techniques with the purpose of finding out which celebrity ones face is most similar to.

1 Related Work

Significant research has been done in the field of facial recognition. Invented in the 1990s, facial recognition has become an integral part of society, from being utilized on cellphones, venues, shopping centers, airports, and by law enforcement, to name a few. Due to the high importance facial recognition holds in society, we wanted to do a project around this task, but wanted to implement a project that has not been done and tested extensively. Coupled with our interest in social media, we decided to utilize facial recognition techniques to identify celebrity lookalikes. In this particular realm, machine learning has been used with celebrities to guess the age or gender of an individual [4], but not to generate an image containing an entire celebrity look-alike. This gap is where we felt we could provide a significant contribution.

2 Data

When searching for our dataset, we prioritized datasets large in size with a diversity of pictures, to prevent any false predictions that may occur by a lack of representation. This project is fascinating in that it has a scope that can extend beyond capturing accurate celebrity look-alikes.

We used the IMDB-WIKI dataset in our implementation. This dataset contains 524,230 images with gender and age labels attached to every picture. The dataset is comprised of a combination of data from the most popular 100,000 actors as listed on the IMDB website [1] and images from Wikipedia 2. When extracting these images from IMDB [1], the timestamp of which the photo was taken was removed, and images with multiple high scored face detections were removed. We assume these images were removed because these faces were more ambiguous and would make the model predict with less precision and accuracy. After sifting through the most relevant images, a total of 461,871 face images from celebrities. Finding the pictures to include in the final dataset were filtered in the same way from the IMDB website as from Wikipedia 2. As a result, a total of 62,359 images were extracted from Wikipedia 2. The following table summarizes the IMDB-Wiki Dataset.

IMDB-WIKI	IMDB	Wikipedia	IMDB-WIKI used for training
524,230	461,871	62,359	260,282 images

Some of the images (especially from IMDB) contain several people. In this case, we only use the photos where the second strongest face detection is below a threshold. However, in order to minimize the amount of bias, we equalize the age distribution. This is done be randomly ignoring images that contain the most frequent ages (in general celebrities are around the ages 20-30, so we wanted to minimize the frequency by which celebrities in this age bracket were prevalent in our training set). As a result, we were left with 260, 282 training images to be inputted in our CNN. This proved to be particularly helpful in our implementation of our model because we were able to produce the celebrity look-alikes of people in various age groups which allowed for more diversity, and consequently more accurate results in our model. The dataset includes well known celebrities from countries that include but are not limited to: the United States, Canada, and the Dominican Republic. Examples are provided below:

Figure 1: [from left to right] Natti Natasha, Miguel Pimental, "Drake" Aubrey Graham

3 Architecture and Design

3.1 Hyperparameters

Our hyperparameters include the dropout rate and the number of convolutional layers. Throughout the quarter, we tested dropout rates that ranged from .5 to .8, but after trial and error, the dropout rate in our final implementation was .5. We ended up choosing this because we wanted a probability of 0.5 for retaining the output of each node in a hidden layer. A larger value closer to 1.0, such as 0.8, retains inputs from the visible layer, which we found did not improve the performance of our model.

In our final implementation, we had 16 convolutional layers and 5 pooling layers. We initially had 1 and incremented the number of layers, as typically, more convolutional layers imply better performance. However, since each convolutional layer reduces the number of input features to the fully connected layers, after we got to the 16th layer, the accuracy gain from the additon of more convolutional layers became negligible.

3.2 Facial Recognition

The model we use utilizes a convolutional neural network. Since a convolutional neural network express faces as vector representations, we utilized Oxford's VGG-Face architecture to find vector representation of faces [2].
Having a vector allows for the data to be organized in a format that is easy to loop through and compare relevant values. In the case of recognizing a face, the distance between an individual's different photos should be as close to 0 as possible. In other words, the model checks if the distance is smaller than a defined threshold to recognize a face
In the case of finding a celebrity look alike, the threshold checking step is overlooked, because the model does not aim to recognize the exact face being compared, but to find a face with similar enough features.
A celebrity look-alike is defined as people who have the minimum distance score, rather than having to be a score under a defined threshold. This allows for the fact that the person used as input as the model is to find an individual with as similar features as possible – they do not have to necessarily be a doppleganger to any particular celebrity.

A visual representation of the VGG-Face Architecture is provided in Figure 2 below.

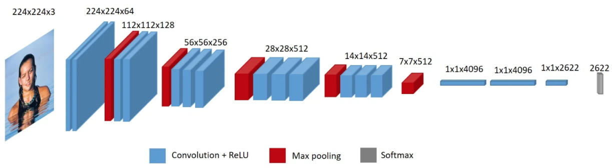

Figure 2: Visualization of VGG-Face Architecture

After representing an image as a vector, we use OpenCVs Haar Cascade classifier in order to detect faces. A Haar Cascade classifier uses the Adaboost learning algorithm to select a small number of important features from a large set to give an efficient result of classifiers [3]. In other words, the algorithm extracts the necessary features from the face that are prevelant in any face. A visualization of the Haar Cascade model is provided in Figure 3 below.

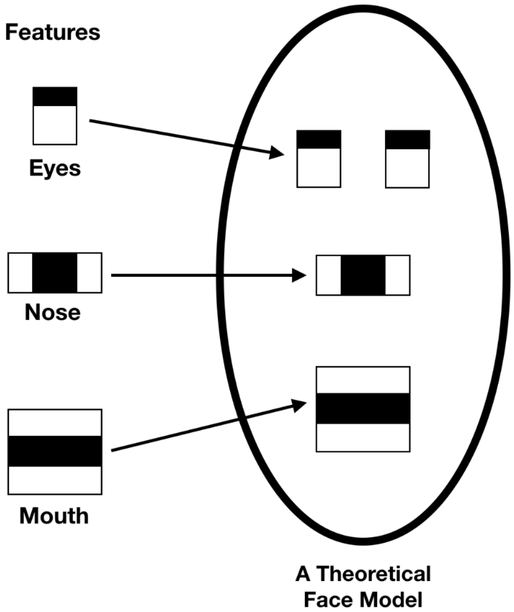

Figure 3: Q Values Gender Breakdown

3.3 Euclidean Distance and Scoring

Both the inputted image and images in IMDB-WIKI data set are represented as 2622 dimensional vectors. Since both have the same dimensionality, it is possible to find the similarity between the inputted image and data set image vector. The first step of the scoring formula is to find the location of all facial features in the image (left eye, right eye, nose, mouth). We then created 2 copies of the image, the first copy was converted to grayscale, and the second copy was run through a sobel filter. This left us with a full color image, to compare skin complection, hair color and eye color, a grayscale image to compare placement of facial features, and a black and white edged image to compare the shapes of facial features. We then compared each of the facial features of the subject to their potential look-alikes by directly subtracting pixels from one another, then taking the sum of the absolute value of the distance. We then add up all of the distances and divide the result by 15 (5 features in 3 different filters) to create a distance score. Faces with lower distance scores are considered to be more similar than faces with higher distance scores. We can represent this as

$$|(\mathbf{S}\odot\mathbf{S})-(\mathbf{O}\odot\mathbf{O})|+\frac{1}{15}\sum_{f=0}^{4}\sum_{i=0}^{img\_width}\sum_{j=0}^{img\_height}|\mathbf{O}_{f,i,j}-\mathbf{R}_{f,i,j}|+|G(\mathbf{O})_{f,i,j}-G(\mathbf{R})_{f,i,j}|+|S(\mathbf{O})_{f,i,j}-S(\mathbf{R})_{f,i,j}|$$

Where the functions *G(x)* and *S(x)* represent the Grayscale and Sobel filters respectively and **O**, **R** represents the original and result images respectively. We also use f to denote the feature number. We then add this distance score to the euclidean distance of the total images.
Computing the Euclidean distance allows the model to have a quantitative value to determine how similar each image in imdb data set to your image. A smaller computed distance implies more similarity exists. We decided to sort the data frame by distance value and pick the 3 most similar celebrity look alikes for a given input. Given this formula, it implies that a larger similarity, or smaller distance is favored.

4 Results

It is difficult to come up with a concretely fail-proof way to measure our results. However, we ultimately decided to use the scoring system, defined in the Euclidean distance section, as a measurement of our algorithm's success. Using this system of measurement, we compared fifty results from the original, milestone implementation to fifty results in our improved implementation, using the same pictures. We then divided each score by the maximum score of all 100 trials in order to get values between 0 and 1, then multiplied this number by 100 to get a score from 0 to 100. From this we compared the average and median distances of both groups. We noticed that the median score improved from 25 in the original implementation to 17 in the new implementation. It is important to remember that a lower score is considered better as it indicates a lower distance, with a distance of of 0 indicating the same face. Out of interest we tried testing the scoring functions on photos of the same person from the same angle and noticed that it came up with a distance of 5, showing that it does have a degree of error, but is a pretty good indicator of similar images. You can see in figure 4 that the program prioritizes facial features and hair color as well as skin complexion. However, it is not necessary to perfectly match in all categories to find your celebrity look alike.

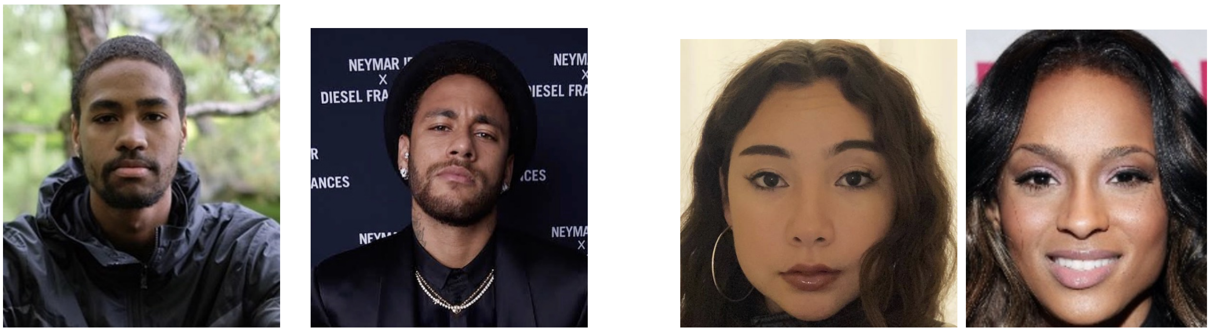

Figure 4: **Left:** Chase's look-alike is Neymar Jr. **Right:** Amanda's look-alike is Ciara

5 Conclusion

Any insights and discussions relevant to the project Our model was able to outperform our baseline model. We had a median score than dropped from 25 to 17, in which a lower number implies a greater similarity, or smaller difference, between the inputted image and celebrity look alike. Overall, we are satisfied with how our model performed, but it still has room to improve. Some possible future steps include increasing the testing set, continuing to increase the dataset to provide even more celebrities to use as reference, or adapt our evaluation metrics to place weight on different features of the face to test what generates more realistic dopplegangers.

References

[1] Ratings and Reviews for New Movies and TV Shows. IMDb, IMDb.com, www.imdb.com/.

[2] erengil, Sefik, et al. Deep Face Recognition with VGG-Face in Keras. Sefik Ilkin Serengil, 15 July 2019, sefiks.com/2018/08/06/deep-face-recognition-with-keras/.

[3] abora, Vince. Face Detection Using OpenCV With Haar Cascade Classifiers. Medium, Becoming Human: Artificial Intelligence Magazine, 4 Feb. 2019, becominghuman.ai/face-detection-using-opencv-with-haar-cascade-classifiers-941dbb25177. . [4] Chen, C. Chen and W. H. Hsu, "Face Recognition and Retrieval Using Cross-Age Reference Coding With Cross-Age Celebrity Dataset," in IEEE Transactions on Multimedia, vol. 17, no. 6, pp. 804-815, June 2015. doi: 10.1109/TMM.2015.2420374