Midterm Paper STOR 390

Ryan Dee

Introduction

Facial Recognition Technology has been around almost 60 years. The first rudimentary tests were done in the 1960's on whether or not "programming computers" could recognize matches between faces by identifying a person's hairline, eyes and nose. The first tests were incredibly unsuccessful. Computers found it easier to beat grandmasters in chess than to match faces to each other. Confounding factors such as lighting, facial expressions and angled photos all impeded the computer from correctly identifying matches. Today systems are much better. Modern facial recognition systems use two algorithms to identify face matches. The initial algorithm maps a person's face and normalizes the data into code, then the second algorithm identifies faces that may be similar. It is important to understand this is different from face detection. Detection is when there is an image that may or may not have faces in it and the algorithm is meant to detect any faces that may or may not be there.

Methods

It is important to understand the mechanism behind a facial recognition algorithm. First there is an image that is used to probe an existing dataset. The image supplied is usually a person of interest and they are looking to be matched against a dataset of standardized faces. However, going from an image to code that a computer can interpret is difficult. This is where the creation of eigenfaces comes from. Eigenfaces are the projection of a standardized pixel image into a 2d vector space. However, this projection means the software can only be used on 2-d images, typically passport photos and police bookings. The projection tracks significant facial features and creates eigenfaces, and can be thought of like a weighted sum of the different values that a person's face is defined by. This is how we go from an image, to the actual data that can go into the support vector machine.

In the data there is a “gallery” which refers to the set of images that are known people. After that the “probe” is the new image that is inserted into the program to verify that there may be a match. This is the two classes that are used, the gallery data and the probe data. Then, the image that we would like to test is measured against the plane that is created by the probe and gallery data by the SVM model. If the image we would like to test is contained within the probe data plane, we have a match.

In this experiment the SVM-based algorithm used 400 frontal images from the FERET database of facial images. The 400 images contained two images from 200 different people. All 400 images were “cleaned”, to remove any background that may interfere with the process of identifying eigenfaces. At the same time, the data was split into 200 training and testing images, where the sets only contained images from one person.

The SVM classifier used a radial kernel and was compared to a PCA-based algorithm, which is the industry standard. The PCA algorithm works by comparing each eigenface to the average eigenface and matching based on the eigenface closest to it when compared to the average. Once processed SVM, and PCA classification were tested on each of the other 200 images in the testing set.

Results

It is an industry standard to test using only the first 30 eigenfaces. In this situation the SVM modeling performed better than the PCA test algorithm. The misclassification rate was 22% compared to 46% for the PCA algorithm. Typically the PCA model does not perform this poorly and the researchers justified this poor performance through difficulty of the images that were selected. This makes sense since a common criticism of the PCA testing method is its lack of broad application because it must be used on very clean images.

Interestingly the researchers also tested the amount of eigenfaces used in the PCA and SVM model. In this case there was a large increase in the correct classification rate for both models for the first 30 eigenfaces, with no statistically significant increase for any after. This s a method to mitigate overfitting which makes many facial recognition systems unreliable.

Moral Dilemma

In terms of moral dilemma this raises interesting questions from the consequentialist, and deontological point of view. This mostly comes from the perspective of selling facial recognition software, and its utilization in areas of the world with people that the data was not trained on. In the same breath, there are consequentialist questions about who gets access to the software. Most of the software that is developed is used to make money and is then commercialized, mostly to governments and law enforcement agencies. This begs the question of whether facial recognition software should be sold to governments that people disagree with. Imagine a situation where facial recognition software is used to find people that were present at a democratic protest in an authoritarian country. Or a situation where the software is sold to a police department that is known to discriminate based on protected classes. From a consequentialist point of view there is a question of "If there is certainty that the government or police agency will get the software anyway, whether or not I sell it to them is irrelevant." At the same time a deontologist would say that you cannot allow it to be a universal maxim to sell recognition software to governments that discriminate.

In another breath, it is important to know that there are people who live in these authoritarian regimes and in the districts of police departments that need protection. It could be that this system is used to find criminals. A consequentialist will have to justify whether they find the consequences of discrimination worth the upside of protecting certain people. Finally, it is worth mentioning this is all determinant on the training data used for the model. If the gallery used is not representative of the probe image the software is more likely to misidentify the image. This is another piece to consider

Bibliography

Phillips, P. Jonathan. “Support Vector Machines Applied to Face Recognition.” *Support Vector Machines for Facial Recognition*, National Institute of Technology, nvlpubs.nist.gov/nistpubs/Legacy/IR/nistir6241.pdf. Accessed 20 Oct. 2024.