Facial Recognition Using PCA

David Huie
Harvey Mudd College

Dustin Rodrigues and **Alex Ruch**
Pomona College

Abstract

Facial recognition is an important area of research of computer science with obvious implications. As one would imagine, it is not a simple task. A large number of variables come into play including facial expressions, hairstyles, camera angle, and lighting. Our paper in particular examines Principal Component Analysis (PCA). This method of facial recogniiton is sensitive to variations of the images. We used databases that contained faces in standard position, with different lighting, and with different face angles, and as expected, our results confirmed that the PCA algorithm performed worse on the images with different lighting and angles.

Introduction

Facial recognition is a varied area of study with applications to security, biometrics, and personal use. In terms of security, one could imagine a situation where an authority has a database of images of faces and security footage of a criminal. It is rarely feasible to manually comb through a database, but a sophisitcated enough facial recogniiton system could possibly determine the identity of the criminal. In biometrics, using a camera to confirm a subject's identity is cheaper than a fingerprint reader or an iris scanner, and is more convenient than having to manually enter a password. The personal uses of facial recognition include training some images of a photo album based on user-based tags and then automatically tagging any new photos with known people.

However, all of these applications have clear deficiencies, most of which come from variations of facial expressions, hairstyles, camera angles, and lighting. These factors contribute to a significant amount of inaccuracy in facial recognition. For personal uses such photo tagging, a wrongly tagged photo is not a very large issue. However, if an authorized user cannot log into their own system because they recently got a haircut, or even worse, an unauthorized user was able to log in, this is a significant security deficiency.

One of the challenges of facial recognition is the extraordinarily large space of possible images. For instance, a 100×100 image will have 10 000 pixels. A naïve method would be to take an image we want classified and compare it against known images using some notion of distance. However, given the aforementioned size of the data, this will be a very computationally intensive task. Instead, Principal Component Analysis (PCA) is used to extract the features with the most amount of variance. Using PCA, one can reduce the space from a dimension of 10 000 to something much more managable, such as 10 or 20 dimensions. Once these principal components are found, we can project an image to be classified onto the principal components and determine which face it most closely resembles.

Algorithm

The PCA Algorithm extracts uncorrelated vectors from vectors that might be correlated; these uncorrelated vectors are called principal components. In more detail, consider a set of mean centered vectors V. That is, the mean vector is calculated from V and is subtracted from each $v_i \in V$. The PCA Algorithm then returns a set E of M eigenvectors e_i such that the quantity $\lambda_i = \frac{1}{M}\sum_{n=1}^{M}(e_i^T v_n)^2$ is maximized. These eigenvectors correspond to the principal components for the v_i's. In practice, the principal components are not found using the equation above, but they are instead extracted from a covariance matrix created from V. The covariance matrix for V would be $C = WW^T$, where W contains the vectors $v \in V$ as column vectors. It is then known that the eigenvalues of C are equal to the e_i's specified above.

In the context of images, the PCA Algorithm would take in as input a matrix W, which is composed of a training set of mean centered images $V = \{v_1, v_2, \ldots v_m\}$ encoded as its column vectors. This is done by creating a one-dimensional vector out of the matrix representation of each image. The set of principal component, or eigenvectors, e_i of the covariance matrix for W represents the training data for a classifier built on PCA. Given an image vector $v_t \notin V$, the image most similar to v_t, $v_i \in V$, is found by first computing $\Omega_i = [d_1 d_2 \ldots d_M]$ with $d_s = e_s^T v_i$ for $i \in \{t\} \cup \{1, 2, \ldots, M\}$. Then, if we minimize $||\Omega_t - \Omega_i||$ for all i, the v_i corresponding to the minimum belongs to the most similar image to v_t. For facial recognition purposes, the person represented by v_t would be identified as the person in the image v_i.

Table 1: Lighting

	Precision	Recall	f1
Face 1			
Face 2			
Face 3			
Face 4			
Face 5			
Face 6			
Face 7			
Face 8			
Face 9			
Face 10			

Table 2: Facial Expressions

	Precision	Recall	f1
Face 1	1.0	1.0	1.0
Face 2	1.0	1.0	1.0
Face 3	1.0	.33	.5
Face 4	1.0	1.0	1.0
Face 5	1.0	1.0	1.0
Face 6	1.0	1.0	1.0
Face 7			
Face 8	1.0	1.0	1.0
Face 9	1.0	1.0	1.0
Face 10	1.0	1.0	1.0
Face 11	1.0	1.0	1.0
Face 12	1.0	1.0	1.0
Face 13	1.0	1.0	1.0
Face 14			
Face 15	.5	1.0	.66

Results

As expected, there was a substantial decrease in performance when images were used that altered lighting or face angles.

Conclusion

As our results showed, facial recognition still has a ways to go. Possible areas for further research include testing how well the algorithm performs on other variations of images. In the real world, people will not all be facing the camera in the same way with the same lighting. Therefore, PCA is not suitable for general-purpose use, but it still can be used in controlled environments. There are other facial recognition systems that can better handle these variations in images, yet even the best ones have some deficiencies, whther it be accuracy, time, or space. Facial recognition is a still-evolving field facing numerous challenges, but it has continued to improve over the years, and hopefully will get better well into the future.