Eigenfaces

Nikita Ramoji

May 2018

Introduction: The History Behind Eigenfaces

For humans, facial recognition is a generally easy process that comes naturally to most people. However, for computers, this task becomes considerably more difficult. The field of computer vision revolves around figuring out how to process and analyze images using computers. One of the most significant topics in this field revolves around facial recognition.

One of the essential problems with writing algorithms for facial recognition is the sheer amount of data involved. A single photo can have hundreds of pixels, and most existing training algorithms require a significant amount of data in order to produce notable results. Due to the computational requirements for even simple computer vision applications, a clear need for a way to represent images using a less data-intensive way became apparent.

In 1987, Larry Sirovich and Michael Kirby, two mathematicians from Brown University, developed an algorithm that allowed for the approximation of essentially any face through a relatively small set of vectors known as eigenfaces. The algorithm essentially functions by representing a set of faces as vectors, determining the mean vector and then calculating the eigenfaces, which are the vectors characterizing the variation of other faces from this mean vector. Eigenfaces proved powerful because instead of having to store an entire database of faces, each face could be produced through a linear combination of the eigenfaces [1].

Model Example

Selecting the Data

We begin with a dataset of faces that we can represent as vectors. We can convert an image of a face into a matrix by viewing it as an $m \times n$ matrix of pixels. A pixel can be represented in many ways, but the techniques in my chosen datasets were to show either the RGB value from 0 to 255 or convert it to some value between 0 and 1, with 0 being completely black and 1 being completely white. For my example, I chose to use datasets provided by sci-kit learn, a popular machine learning library. I chose the Olivetti Faces and the Labeled Faces in the Wild datasets to work with [2].

Although it is not necessarily relevant for understanding the mathematics behind this model example, I also want to emphasize the importance of selecting a diverse and representative data set when determining eigenfaces in the real world. The idea behind eigenfaces is that by determining the most significant ways that faces differ from each other, we can then generate any face. However, if the original data set of faces that we base this on fails to properly represent specific ethnic groups or genders, then the mean face we calculate will clearly differ from what the actual mean face from the population should be. Additionally, the actual faces found in the population are likely to have features not found in the original data set. A significant problem with data analysis in general in computer science right now is that many of the existing databases prioritize white, male faces. Since existing algorithms work off of these data sets, modern facial recognition is notably worse for people of color and is worst on women of color [3].

Calculating the Eigenvectors

Before beginning my model example, I would like to note that I took my main understanding of the algorithm behind eigenfaces from Jeremy Kun's informative blog on the subject [4], but all of the equations and explanations of linear algebra concepts were taken from our course textbook [5].

Now that we have represented all of the images as matrices, we can use linear algebra to obtain the eigenvectors for the set of faces. We will refer to these vectors as eigenfaces. The idea behind this mode of facial recognition is that all faces can be viewed as a linear combination of a set of eigenfaces. Therefore, if we can determine this set of eigenfaces it will become far easier to detect and recognize faces within images.

We will first transform each face into a vector by concatenating all of the rows of pixels in the dataset's matrix so that we can represent each image as a vector with $m \times n$ elements. We will use these vectors as the columns of our matrix A.

We then want to subtract the vector representation of the mean face from all of the other faces. We can calculate the mean vector m using this formula:

$$m = \frac{o^T A}{o^T o},$$

where o is a column vector with as many 1's as the matrix A has rows. You then subtract m from each column in A [6].

This step of normalizing the matrices makes it easier to then obtain the covariance matrix C. Each element c_{ij} in the covariance matrix can be calculated by taking the dot product of the ith column and the jth column in the A matrix. Therefore, this matrix can be calculated as $C = AA^T$. The eigenvectors of this covariance matrix are the eigenfaces that we have been trying to compute. However, in practice the singular value decomposition (SVD) of A can be used instead of calculating the covariance matrix [7].

Singular Value Decomposition

Diagonalization is a useful linear algebra technique, but it doesn't work on all matrices. Singular value decomposition can be used on any $m \times n$ matrix A, including those that can't be factored as $A = PDP^{-1}$. For any $m \times n$ matrix A with rank r, there exists a singular value decomposition such that

$$A = U\Sigma V^T,$$

where the diagonal entries of Σ are the singular values of A and where U, V are orthogonal matrices. The columns of U are the left singular vectors of A and the columns of V are the right singular vectors.

As stated in our textbook, the singular values of A are equivalent to the eigenvalues of $A^T A$ arranged in decreasing order. We know that

$$C = AA^T,$$

so through simple substitution we know that

$$C = (U\Sigma V^T)(U\Sigma V^T)^T.$$

Additionally, for any product of matrices, $(AB)^T = B^T A^T$, implying that

$$C = (U\Sigma V^T)(V\Sigma^T U^T).$$

Since V is an orthogonal matrix, $V^T V = I$, implying that

$$C = U\Sigma I\Sigma^T U^T = U\Sigma\Sigma^T U^T$$

Now, let $D = \Sigma\Sigma^T$. Therefore, we now have

$$C = UDU^T,$$

and since U is an orthogonal matrix then $U^T = U^{-1}$ and

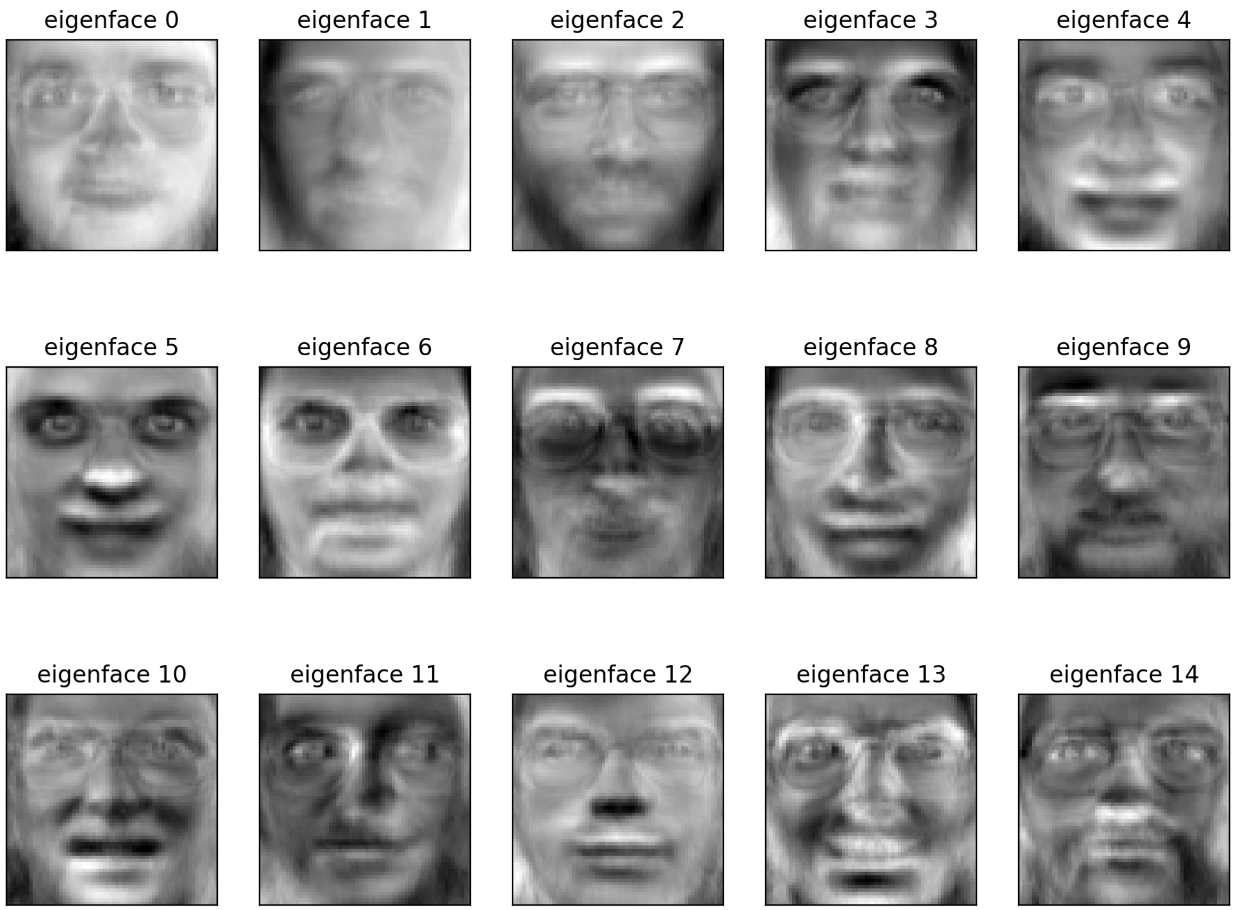
$$C = UDU^{-1},$$

where the vectors of U are the eigenfaces (essentially the eigenvectors) that we have been looking for!

Now that we have found all of the eigenvectors, we can see how any face, represented as a column in A, varies from the mean face, represented as the mean vector m calculated above. However, retaining all of the eigenfaces that we just found is computationally expensive and unnecessary for most uses of eigenfaces. In most cases, we would want to focus on the k eigenfaces with the largest corresponding eigenvalues.

The Results

Upon doing this with the Olivetti Faces dataset, we obtained these eigenfaces:



And upon doing this with the Labeled Faces in the Wild dataset, these eigenfaces were procured:

Why Are Eigenfaces Important

Data Compression

As stated before, one of the biggest achievements of eigenfaces is the ability to generate nearly any face using a linear combination of the eigenfaces. The ability to generate faces in this way makes it considerably easier to store a large number of images because you merely need to store the original set of eigenfaces and then the linear combinations needed to generate the face. In fact, according to the original paper, less than 100 eigenfaces are required to generally produce a reasonable representation of an image [1].

Facial Recognition

The primary use of eigenfaces is for facial recognition. Current science believes that facial recognition in humans is not necessarily based on any specific distinct feature but based on the overall combination of features that comprise the face. This theory also explains why humans are prone to seeing ”faces” in objects or nature even when there is no actual face [8].

Facial recognition works on the theory that every face is disinct from other faces. Therefore, by considering the unique features of a face you can distinguish it with reasonable accuracy by projecting the corresponding vector into the eigenspace generated by all of the eigenfaces. By determining the eigenfaces that most closely resemble the face we are trying to identify, with a smaller Euclidean distance suggesting a closer resemblance, we can either identify the face or determine that the distance is too large and the person is not recognized [9].

The use of facial recognition has both benefitted and harmed society. For security purposes it is essential. From secure buildings that should only be accessed by select individuals to the average citizen being able to protect their smartphone from being unlocked by strangers, facial recognition allows a way to personalize security such that only specific individuals can access key information, locations or items.

On the other hand, this same technology can be used for government surveillance and potential infringiment on individuals' privacy. Skynet, a national Chinese surveillance program, is used to catch and predict crimes [10]. However, there exists a significant amount of controversy over whether or not it is ethical to use computer algorithms to determine whether or not a person is likely to become a criminal. Fundamentally, computer algorithms often work based off the data they are given and the mindset of the person designing them, including any implicit biases that may be included. For instance, there is evidence that people of color are often more likely to be faced with police brutality or unfairly punished for a crime that they did not commit [11]. The use of an algorithm to determine whether or not someone is likely to become a criminal gives a sense of legitimacy to an accusation that may be baseless. Furthermore, there are studies evidencing that police uses of facial recognition rarely test the accuracy of the findings, which only further strengthens the importance of setting ethical limits on the use of facial recognition technology [12].

Why Eigenfaces Matters to Me

I must admit, as someone dipping their toes into the field of computer science currently, being able to see a real-world application of linear algebra being used in an algorithm is inspiring to me. I enjoy the idea of being able to apply the skills and understanding of mathematics that I've picked up in this class to my future career. I also find it incredible how a reasonably simple application of linear algebra fundamentals is so powerful. Eigenfaces have a tremendous number of applications, yet the actual mathematics behind them was simple enough that I, a person who has only been studying these concepts for the duration of a single semester, was able to both understand and apply them myself. I also must admit that the fact that the mathematicians behind the eigenfaces algorithm, Sirovich and Kirby, are alumni of Brown University makes me feel like I, and my fellow students surrounding me, could also potentially develop these kinds of algorithms in the future. Thank you for everything you have taught me over this semester, and I hope to continue exploring the applications of eigenfaces and linear algebra in the future.

References

[1] L. Sirovich and M. Kirby. "Low-dimensional procedure for the characterization of human faces". *J. Opt. Soc. Am. A* 4.3 (Mar. 1987), pp. 519–524. DOI: 10.1364/JOSAA.4.000519. URL: josaa.osa.org/abstract.cfm?URI=josaa-4-3-519.

[2] Scikit-Learn Project. *Dataset Loading Utilities*. URL: scikit-learn.org/stable/datasets/index.html.

[3] Tom Simonte. "Photo Algorithms ID White Men Fine—Black Women, Not So Much". *Wired* (Feb. 6, 2018). URL: https://www.wired.com/story/photo-algorithms-id-white-men-fineblack-women-not-so-much/.

[4] Jeremy Kun. *Eigenfaces, for Facial Recognition*. URL: https://jeremykun.com/2011/07/27/eigenfaces.

[5] David C. Lay, Steven R. Lay, and Judi J. McDonald. *Linear Algebra and its Applications*.

[6] Massachusetts Institute of Technology. *Principal Component Analysis*. Sept. 7, 2017. URL: web.mit.edu/18.06/www/Spring17/Statistics-and-PCA.pdf.

[7] Bhiksha Raj. *Assignment 2: Face Detection*. URL: http://www.cs.cmu.edu/~pmuthuku/mlsp_page/assignments/assignment2_hints.html.

[8] Elizabeth Svoboda. "Faces, Faces Everywhere". *New York Times* (Feb. 13, 2007). URL: www.nytimes.com/2007/02/13/health/psychology/13face.html.

[9] Müge Çarıkçı and Figen Özen. "A Face Recognition System Based on Eigenfaces Method". *Procedia Technology* 1 (2012). First World Conference on Innovation and Computer Sciences (INSODE 2011), pp. 118–123. ISSN: 2212-0173. URL: www.sciencedirect.com/science/article/pii/S2212017312000242.

[10] Tara Francis Chan. "Parts of China are using facial recognition technology that can scan the country's entire population in one second". *Business Insider* (Mar. 26, 2018).

[11] Kia Makarechi. "What The Data Really Says About Police and Racial Bias". *Vanity Fair* (July 14, 2016).

[12] Mike Loukides. "The Ethics of Face Recognition". *O'Reilly Media* (Dec. 13, 2016).