MTH 261 Paper: Facial Recognition

Pasang Sherpa and Eleanor Quirk

Portland Community College

February 27, 2020

Abstract

Facial recognition is an algorithm that can be used to match a face from a given image or video with a face from a database. By matching the faces, the identity of the person whose face is processed can often become known. The ability to identify a face is in high demand for a spectrum of uses, including user identification and authentication, law enforcement, attendance tracking, and social media streamlining. Using facial recognition algorithms can have many benefits, though it also raises many questions on ethics and security that must be addressed before it is implemented on a large scale [`?youtube:howdoes`]. While the implementation of facial recognition algorithms is socially complicated, the mathematical process of facial recognition is complex in its own regard, requiring several steps and matrix calculations. First, a database of faces must be collected, and their eigenfaces calculated. Next, an inputted image must be compared to those eigenfaces, and the weight of each eigenface on the inputted face must be determined. Finally, it must be calculated whether or not the inputted image is a face, and if so, whether it matches or is close to any of the faces in the database [`?whosthatface:explanation`]. This algorithm uses many key elements of linear algebra, as well as Principal Component Analysis (PCA), in order to accurately match a given face with a known face.

1 Introduction

Facial recognition is an easy task for humans. Even from a young age, humans are able to distinguish between faces and identify what face corresponds to which person [`?iot:howworks`]. This is extremely useful, as it enables us to quickly and accurately determine who we are speaking with, who is a friend and who is a foe, who represents us in government, who is responsible for our medical care, who you interact with regularly and who is new to you. Ultimately, facial recognition allows people to recognize one another and build relationships that are essential to our survival as a community-based species.

The importance of facial recognition is immeasurable, so it would be amazing if we were able to massively enhance our ability to recognize faces by using computers. Using computers to recognize faces is a difficult process, but continues to improve as more mathematicians and scientists approach the many problems that arise as a computer attempts to connect a person with a series of pixels [`?iot:howworks`].

Individuals can be identified in many ways using different bio metrics such as fingerprint, iris, voice patterns. Facial recognition is gaining more favor because it requires no physical interaction on behalf of the user. A single individual can be tracked in a massive crowd without them knowing. It is also accurate depending on their false acceptance rate and false rejection rate.

2 How does computation facial recognition work?

One of the most important aspects of computation recognition is creating a faceprint. A faceprint is a collection of important "landmarks" on a face, which create a map of a face. There are dozens of landmarks on human faces, though some of the most important ones for facial recognition include "[d]istance between the eyes, [w]idth of the nose, [d]epth of the eye sockets, [t]he shape of the cheekbones, [and] [t]he length of the jaw line" [`?hsw:facerecognition`]. One of the first steps in facial recognition is to use software to collect data on the landmarks of a face in an image, and to create a faceprint based on that data [`?hsw:facerecognition`][`?iot:howworks`].

The system compares the individual's landmark to the individual they claim to be, or to all the individuals in the database to identify that particular person. Depending on the precisely calculated measurement, it gives us the list of matches found. When choosing a facial recognition system software, we need to pay attention to the false acceptance rate, which means an unauthorized person will not be allowed access and the false recognition rate, means that the authorized person will have to try several times to get access. So, it is good to have a very low FAR and low FRR for accuracy.

3 The math behind facial recognition

The first step of the facial recognition algorithm is to prepare a database of faces, which will be compared with an inputted face [`?whosthatface:explanation`]. The goal is to determine whether the inputted face (new, unknown face) is the same face as any of the faces in our database (known faces), or if the inputted face does not match any of the faces in our database.

Note: All the images involved with this algorithm should have the same size in order for the calculations to make sense. Ideally, all the images should be of just faces (no bodies, hands, obstructive accessories, etc.), and should all be taken at similar angles and under similar lighting conditions.

3.1 Preparing the Known Faces

An image is an array of pixels. "A pixel is the smallest unit of a digital image or graphic that can be displayed," and is essentially a tiny square that is a particular color [`?techopedia:pixel`]. When many pixels are arranged, they come together almost like a jigsaw puzzle, in order to create an image. The shape that images almost always take on is a rectangle, which makes it fairly easy for images to be represented in matrices, where each pixel is represented by an entry in the matrix, and its position in the image corresponds to its position in the matrix. If the image is grayscale (rather than colored), only one number (the intensity of the gray at a particular point) is needed to represent a pixel. Thus, a grayscale image can be easily represented by a matrix of numbers.

In calculating eigenfaces, it is necessary to work with vectors rather than matrices. To turn the matrices (representing images of faces) into vectors while maintaining all the data they contain, stack all the columns of a matrix into one long column vector. For example, we can turn a matrix M, representing intensity of grayness in an image, into a long column:

$$M = \begin{bmatrix} 33 & 205 & 176 & 40 & 213 \\ 50 & 188 & 93 & 51 & 77 \\ 64 & 28 & 12 & 30 & 195 \\ 25 & 192 & 28 & 195 & 38 \\ 57 & 37 & 49 & 29 & 41 \end{bmatrix} = \begin{bmatrix} \mathbf{a_1} & \mathbf{a_2} & \mathbf{a_3} & \mathbf{a_4} & \mathbf{a_5} \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{a_1} \\ \mathbf{a_2} \\ \mathbf{a_3} \\ \mathbf{a_4} \\ \mathbf{a_5} \end{bmatrix} = \begin{bmatrix} 33 \\ 50 \\ 64 \\ 25 \\ 57 \\ 205 \\ 188 \\ 28 \\ 192 \\ 37 \\ 176 \\ 93 \\ 12 \\ \vdots \end{bmatrix}$$

All the images in the database of known faces must be transformed into vectors by this method, so that each known face image is a long column vector:

$$\text{Face 1: } \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ \vdots \end{bmatrix}, \text{ Face 2: } \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ \vdots \end{bmatrix}, \text{ Face 3: } \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ \vdots \end{bmatrix}, \text{ Face 4: } \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \\ \vdots \end{bmatrix}, \text{ and so on...}$$

Now we must find the average face. The average face is the vector that results when the entries in each row are added, and the resulting value is divided by the total number of known faces. This averaging is done for each row of the known face vectors, to produce a vector that is the same length as the known face vectors.

$$\text{Average face} = \left(\tfrac{1}{\text{total faces}}\right) \begin{bmatrix} a_1 & + & b_1 & + & c_1 & + & d_1 & + & \cdots \\ a_2 & + & b_2 & + & c_2 & + & d_2 & + & \cdots \\ a_3 & + & b_3 & + & c_3 & + & d_3 & + & \cdots \\ a_4 & + & b_4 & + & c_4 & + & d_4 & + & \cdots \\ \vdots & + & \vdots & + & \vdots & + & \vdots & + & \ddots \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ \vdots \end{bmatrix}$$

We next subtract the average face vector from each known face vector, to produce altered known face vectors. These altered known face vectors must then be combined to form the columns of a matrix, which we will call matrix A:

$$\text{Altered Face 1} = \mathbf{f_1} = \begin{bmatrix} a_1 - m_1 \\ a_2 - m_2 \\ a_3 - m_3 \\ a_4 - m_4 \\ \vdots \end{bmatrix}, \text{ Altered Face 2} = \mathbf{f_2} = \begin{bmatrix} b_1 - m_1 \\ b_2 - m_2 \\ b_3 - m_3 \\ b_4 - m_4 \\ \vdots \end{bmatrix}, \text{ and so on...}$$

$$A = \begin{bmatrix} \mathbf{f_1} & \mathbf{f_2} & \mathbf{f_3} & \mathbf{f_4} & \cdots \end{bmatrix}$$

Note that this matrix A will be quite large. There is a row for every pixel in an original image, and a column for every known image, which means that there will likely be thousands of rows in A, and dozens or hundreds of columns. The size of matrix A becomes problematic in the next step of the mathematics involved with facial recognition, as we begin to perform more complicated calculations such as matrix multiplication and determination of eigenvectors and their associated eigenvalues.

3.2 Calculating Eigenfaces

Eigenfaces are "the eigenvectors of the covariance matrix of the set of [known] face images" [`?scholarpedia:eigenf`
We use eigenfaces to describe a new face in terms of known faces; through analysis of how the eigenfaces are similar or dissimilar to the new face, we are able to determine a possible identity for the new face, or see if the new face is entirely unknown. Hence, calculating the eigenfaces of our known faces is imperative to the success of facial recognition algorithms.

To calculate our eigenfaces, we must first find the covariance matrix of our matrix A, the matrix whose columns are vectors representing our altered known face images. The covariance matrix is found via the matrix multiplication AA^T, but this multiplication brings up a dilemma: since A is a matrix with thousands of rows, and A^T is a matrix with thousands of columns, AA^T results in a matrix with thousands of rows and thousands of columns, which is very unwieldy and computationally demanding. Moreover, not all of the information stored in this covariance matrix is useful, meaning that a computer would have to perform many unnecessary calculations, which would significantly slow it down.

To avoid so many cumbersome and unnecessary calculations, we perform the multiplication A^TA rather than AA^T. This results in a matrix that has dozens (or hundreds) of rows and columns, and is much more manageable than the the matrix found by AA^T.

Next, we find the eigenvectors of the matrix formed by A^TA, and calculate the corresponding eigenvalues. "The eigenface images calculated from the eigenvectors of A^TA span a basis set with which to describe face images," which means that a new face can be written as a linear combination of the eigenfaces found through the above series of calculations. [`?eigenfaces:turk`] That is,

$$\text{new face} = c_1[\text{eigenface}_1] + c_2[\text{eigenface}_2] + c_3[\text{eigenface}_3] + \cdots$$

where $c_1, c_2, c_3, \ldots$ are scalars. From this linear combination, we can see that if any particular scalar was to be large in comparison to the other scalars, then the eigenface associated with that scalar would "contribute" more to the new face than any other eigenface. In other words, the new face would be mostly composed of that particular eigenface. As a result, the identity of the new face is most likely to match the identity of the known face whose eigenface has the largest associated scalar. If no scalar is very prominent, this means that the new face is unknown to the data set; the new face does not match any of our known faces.

3.3 Principal Component Analysis

PCA is a statistical method, which emphasizes obtaining a smaller number of uncorrelated variables from a vast number of variables found in a data set [`?PCA:wiki`]. The dimensionality reduction reduces the number of data sets, while preserving as much information as possible. It will reduce the data set from a two-dimensional plane to a one or if we had a three-dimensional plane to two or one line. This technique aims to find some hyper plane, to project the points onto. Projection means to map a set into a subset. To find this hyper plane, we need to know what are

variance, co variance and co variance matrix. "Variance measures the variation of a single random variable, whereas co variance is a measure of how much two random variables vary together" [?Covariancematrix:datascience].
"Co variance matrix, or variance–co variance matrix, is a matrix whose element in the i, j position is the co variance between the i-th and j-th elements of a random vector" [?covariancematrix:wiki]. For example, a 3×3 covariance matrix is formed by the variances and the covariance of this data set, the diagonal entries of this covariance matrix are the variances.

$$\begin{bmatrix} Var_1 & Cov_{1,2} & Cov_{1,3} \\ Cov_{2,1} & Var_2 & Cov_{2,3} \\ Cov_{3,1} & Cov_{3,2} & Var_3 \end{bmatrix}$$

Let us assume that the given data set below is a measured variables of various faces. The given table data is from a site. It has been used here as a facial measurements. [?calculatingcovariance:jamesmccaffrey]

Facial landmark data set			
	Distance of the eye	width of the nose	length of the jawline
Face 1	64.0	580.0	29.0
Face 2	66.0	570.0	33.0
Face 3	68.0	590.0	37.0
Face 4	69.0	660.0	46.0
Face 5	73.0	600.0	55.0

Variance formula: $\sigma^2 = \frac{\sum_{i=1}^{n}(x_i-\mu)^2}{n}$ Covariance formula: $cov_{x,y} = \frac{\sum_{i=1}^{N}(x_i-\bar{x})(y_i-\bar{y})}{N-1}$

In the given table below is the computed values of variance and covariance of the data set:

variance and covariance	
var(X)	11.50
var(Y)	1250.00
var(Z)	110.00
covar(XY)	50.00
covar(XZ)	34.75
covar(YZ)	205.00

Covariance matrix of this data set:

$$A = \begin{bmatrix} 11.50 & 50.00 & 34.75 \\ 50.00 & 1250.00 & 205.00 \\ 34.75 & 205.00 & 110.00 \end{bmatrix}$$

Now we compute the eigenvalues and eigenvector of the covariance matrix:
Here, $A = \begin{bmatrix} 11.50 & 50.00 & 34.75 \\ 50.00 & 1250.00 & 205.00 \\ 34.75 & 205.00 & 110.00 \end{bmatrix}$
We know λ is and eigenvalue of A

$$Av = \lambda v, \text{ for some nonzero } v$$
$$(A - \lambda I)v = 0,$$

$$\begin{bmatrix} 11.50 & 50.00 & 34.75 \\ 50.00 & 1250.00 & 205.00 \\ 34.75 & 205.00 & 110.00 \end{bmatrix} - \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$$

$$(A - \lambda I) = \begin{bmatrix} 11.50 - \lambda & 50.00 & 34.75 \\ 50.00 & 1250.00 - \lambda & 205.00 \\ 34.75 & 205.00 & 110.00 - \lambda \end{bmatrix}$$

$$\det(A - \lambda I) = \begin{vmatrix} 11.50 - \lambda & 50.00 & 34.75 \\ 50.00 & 1250.00 - \lambda & 205.00 \\ 34.75 & 205.00 & 110.00 - \lambda \end{vmatrix}$$

According to wolframalpha the eigenvalues of the covariance matrix are:

$$\lambda_1 \approx 1288.13$$

$$\lambda_2 \approx 83.1249$$

$$\lambda_3 \approx 0.241738$$

The eigenvector of the covariance matrix are:

$$v_1 \approx \begin{bmatrix} 0.25064 \\ 5.70451 \\ 1 \end{bmatrix}, v_2 \approx \begin{bmatrix} 0.351996 \\ -0.190766 \\ 1 \end{bmatrix}, v_3 \approx \begin{bmatrix} -2.86766 \\ -0.0493932 \\ 1 \end{bmatrix}$$

The covariance matrix of a data is calculated, and the most significant eigenvectors of the covariance matrix are selected. The long axis that maximally spreads our data sets is where our data gets projected onto. Through this method of selecting the largest eigenvectors of the covariance matrix, we can reduce the data to a lower dimension by projecting onto those eigenvectors. For facial recognition, it is much easier to match and differentiate faces when the data set is spread out.

Here we have three eigenvectors/values of the data set. Two of the eigenvalues are the largest eigenvalues and one of the eigenvalues is zero. We will discard the third eigenvector, which has an eigenvalue of zero. Than rearrange our axis along the eigenvectors. Then the data set will be projected onto the two eigenvectors with the highest eigenvalues. Representing the data in two dimension instead of three helps simplify the data and it is easier to visualize the data set.

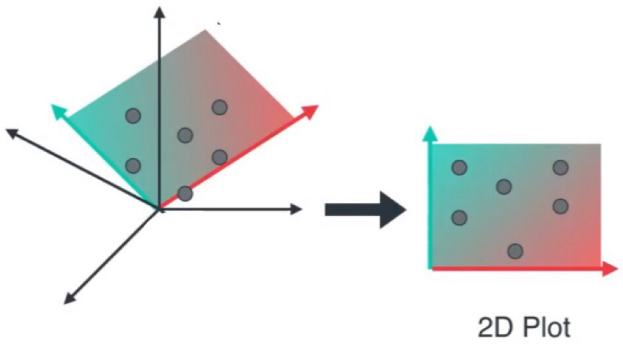

Fig 1:2D

4 How facial recognition has been used, and could be used

Facial recognition is not a new thing. One of the pioneers of automated face recognition Woody Bledsoe's initial approach involved the manual marking of various landmarks on the face such as the eye centers, mouth, etc., and these were mathematically rotated by computer to compensate for pose variation [?historyofinfosecurity:Acomprhensiveha]. The distances between landmarks were also automatically computed and compared between images to determine identity [?historyofinfosecurity:Acomprhensiveha]. With the advancement of technology and information, different computer algorithms can identify individuals from a crowd quickly without passers-by even being aware of the system.

Facial recognition technology can be used in a vast number of situations:

- **Criminal Detection:** In airports, large events, monuments, and other areas, facial recognition can be used in conjunction with security cameras to scan crowds and identify "known troublemakers," so that security teams can be more informed and better prepared to manage situations that might come up [?youtube:howdoes].

- **User Identification:** Users of Apple's iPhone X can now unlock their phone using "Face ID" rather than "Touch ID" or a passcode. This is a very secure method of unlocking a phone, as there is only a "one-in-a-million chance" that someone else would be able to unlock the phone with their face, and the user's eyes must be open in order to unlock the phone (thus preventing someone else from unlocking it while the user is sleeping) [?lifewire:whatis]. Moreover, Mastercard is looking into how facial recognition could be used as a way to authenticate a purchase made with a credit card [?youtube:howdoes].

- **Social Media:** Social media platforms like FaceBook can use facial recognition software to identify and tag people in a picture. This would save FaceBook users time because they would no longer need to manually tag people in the picture. This automatic tagging would help users find images and connect with their online friends.

- **Attendance:** Some colleges have begun to use facial recognition technology to take attendance, and prevent cheating. Some places of worship have also used facial recognition to track who comes and how often; this allows the staff of the place of worship to write donation requests in more targeted and effective ways [?iot:howworks].

- **Reduce Toilet Paper Theft and Waste:** In an effort to reduce toilet paper waste in parts of China, facial recognition systems have been combined with toilet paper dispensers in public restrooms. A restroom user must show their face to the system in order to receive a certain amount of toilet paper from the dispenser; the dispenser will then refuse to provide more toilet paper to the same person for several minutes.

These are just a few examples of how facial recognition technology is currently being used. As this technology continues to increase in accuracy and efficiency, facial recognition will undoubtedly be used in more and more situations, and will almost certainly become an important part of many aspects of our lives.

5 Issues arising from use of facial recognition

As with any new, powerful, and versatile technology, facial recognition brings up several concerns whenever it is put to use.

One of the main concerns about facial recognition is its potential to dramatically decrease individual privacy. For example, if an individual's picture is posted on a social media platform and facial recognition software identifies that individual, that person's name is now attached to that image, perhaps irrevocably. This could be problematic for that person for a variety of reasons, some examples of which are listed here:

- They didn't want their name associated with that particular picture. (Maybe the picture is of them making a silly, embarrassing face, that they didn't know would be posted.)
- The picture reveals something about their location, and having their name attached to it could be dangerous for them. (Perhaps the person is trying to escape an abusive environment or stalker; having their name associated with a location could help the abuser or stalker find them.)
- Having their name attached to that picture could impact how others treat that person. (If the picture shows the person at a Pride Parade, and some of the person's relatives are not accepting of the LGBTQ+ community, then the person may face discrimination at home or family gatherings.)

These are just three examples of how facial recognition could decrease a person's privacy online, and might reveal information that the person would rather keep private. Besides decreasing online privacy, facial recognition could also impair privacy in real life. For example, if facial recognition is used to observe trends on attendance in places of worship, information about how many times an individual has gone to such a place becomes a data point that anyone could potentially access without the person's knowledge or consent.

There is also a concern that facial recognition algorithms could make it easier for organizations to profile individuals, and to give them targeted ads. As an example, a supermarket in the US is working on what they call "smart shelves," where cameras in the aisles use facial recognition software to identify a shopper's age, gender, and other information about them. That information is then used to anticipate what the shopper might want, and to decide what ads and information the shopper sees [`?youtube:howdoes`]. This use for facial recognition is touchy, because it makes assumptions about a person based on their appearance, thus making use of and enforcing stereotypes.

Facial recognition also raises issues because of its potential for inaccuracy. A particular algorithm may be unable to identify a person when it should be able to, or may make an incorrect identification, and both of these could lead to significant problems, particularly in high-stakes situations such as law enforcement. According to Greg Foot, some shopkeepers are using facial recognition to identify local thieves, and to "alert them when shoplifters enter the store" [`?youtube:howdoes`]. What happens next is up to the shopkeeper, but let us imagine that they decide to call the police. What if the person who enters the store is not a shoplifter, but rather happens to look like one that the facial recognition software has in its database? Then, when the police respond to the shopkeeper's call, the shoplifter-lookalike will have a surprising and stressful interaction with them, and the police's time and energy will have been wasted. Thus, in law enforcement and other situations, where it is very important to match identities correctly, the potential for facial recognition software to incorrectly match names and faces is concerning.

6 Conclusion

Facial recognition algorithms are very powerful, and have many uses both in the present day and in the future. The mathematical process of getting a computer to recognize a face is complex and

requires many time-consuming calculations. Fortunately, PCA and other mathematical maneuvers reduce the dimensions of the data involved with facial recognition algorithms, thereby making them much more efficient. This increased efficiency makes these algorithms usable for a wide variety of applications, spanning from attendance-taking to criminal investigations. Because facial recognition can be used in so many ways, its use has become a topic of discussion, and several issues (including ethical dilemmas) have been identified as arising from its use. These issues must be discussed and addressed, as facial recognition technology becomes increasingly fast, accurate, and widespread.