Drexel University

Home
Tutorials
Contact info
Tasks

Eigenface Tutorial

Keywords: Matlab, Face Recognition, Eigenfaces, Computer vision

This picture represents the set of images used to create our eigen space for face recognition. The goal of this tutorial is to apply the eigenface approach to recognize someone's face. The overall problem is to be able to accurately recognize a person's identity and take some action based on the outcome of the recognition process. Recognize a person's identity is important mainly for security reason, but it could also be used to obtain quick access to medical, criminal, or any type of records. Solving this problem is important because it could allow personnel to take preventive action action, provide better service - in the case of a doctors appointment, or allow a person access to a secure area. This tutorial will shows you how to build a face recognition program using Matlab.

Motivation and Audience

This tutorial has been developed to get people interest on computer vision and face recognition; in addition to give you some guidance on how to approach the problem. The reader should have the following background and interests:

- ***Moderate Matlab knowledge***
- ***Being familiar with Linear Algebra concepts***
- ***Desire to learn about face recognition using Matlab***
- ***In addition, would like to learn more about computer vision***

The rest of the tutorial is presented as follows:

Parts List and Sources

US-based vendors to obtain material to complete this tutorial include
Mathworks

To complete this tutorial, you'll need the following items

Software	VENDOR	Version	PRICE	QTY
Matlab	Mathworks	6.1	$500	1

Construction

This section gives step-by-step instructions along with photos and formulas on how to recognize faces and implemented into Matlab. All the necessary files to complete this tutorial would be provided.

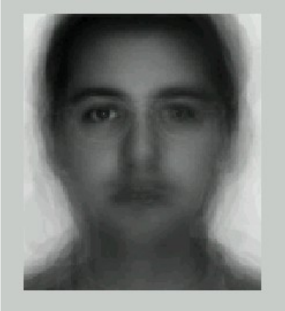
Steps

1. The first step is to obtain a set S with M face images. In our example M = 25 as shown at the beginning of the tutorial. Each image is transformed into a vector of size N and placed into the set.

$$S = \{\Gamma_1, \Gamma_2, \Gamma_3, \ldots\ldots\ldots, \Gamma_M\}$$

2. After you have obtained your set, you will obtain the mean image $\boldsymbol{\Psi}$

$$\Psi = \frac{1}{M} \sum_{n=1}^{M} \Gamma_n$$



3. Then you will find the difference $\boldsymbol{\Phi}$ between the input image and the mean image

$$\Phi_i = \Gamma_i - \Psi$$

4. Next we seek a set of M orthonormal vectors, $\mathbf{u}_\mathbf{n}$, which best describes the distribution of the data. The k^th vector, $\mathbf{u}_k$, is chosen such that

$$\lambda_k = \frac{1}{M} \sum_{n=1}^{M} \left(u_k^T \Phi_n\right)^2$$

is a maximum, subject to

$$u_l^T u_k = \delta_{lk} = \begin{cases} 1 & \text{if } l = k \\ 0 & \text{otherwise} \end{cases}$$

Note: $\mathbf{u}_k$ and $\boldsymbol{\lambda}_k$ are the eigenvectors and eigenvalues of the covariance matrix **C**

5. We obtain the covariance matrix **C** in the following manner

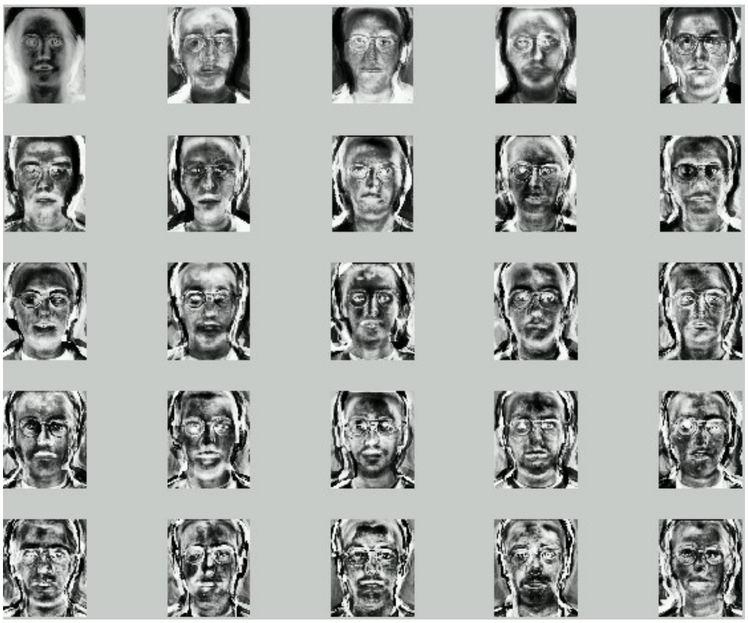
$$C = \frac{1}{M}\sum_{n=1}^{M} \Phi_n \Phi_n^T$$
$$= AA^T$$
$$A = \{\Phi_1, \Phi_2, \Phi_3, \ldots\ldots, \Phi_n\}$$

6. A^T

$$L_{mn} = \Phi_m^T \Phi_n$$

7. Once we have found the eigenvectors, $\mathbf{v_l}$, $\mathbf{u_l}$

$$u_l = \sum_{k=1}^{M} v_{lk}\Phi_k \qquad l = 1, \ldots\ldots, M$$



These are the eigenfaces of our set of original images

Recognition Procedure

1. A new face is transformed into its eigenface components. First we compare our input image with our mean image and multiply their difference with each eigenvector of the L matrix. Each value would represent a weight and would be saved on a vector Ω.

$$\omega_k = u_k^T(\Gamma - \Psi) \quad \Omega^T = [\omega_1, \omega_2, \ldots\ldots, \omega_M]$$

2. We now determine which face class provides the best description for the input image. This is done by minimizing the Euclidean distance

$$\varepsilon_k = \|\Omega - \Omega_k\|^2$$

3. The input face is consider to belong to a class if ε_k is bellow an established threshold θ_ε. Then the face image is considered to be a known face. If the difference is above the given threshold, but bellow a second threshold, the image can be determined as a unknown face. If the input image is above these two thresholds, the image is determined NOT to be a face.

4. If the image is found to be an unknown face, you could decide whether or not you want to add the image to your training set for future recognitions. You would have to repeat steps 1 trough 7 to incorporate this new face image.

See Results

The source code face recognition using Matlab is provided below:

To be compiled with Matlab Editor/Debugger
Note: download the Face Recognition Kit rather than cutting and pasting from below.
Use WinZip to unzip the files. All the necessary files are included.

Example of Face.m script for face recognition using Matlab:

Click here to go to the code

Download Face Recognition Kit

If you follow the steps on the tutorial you will obtain the following results. These results are different from the ones shown above, since the images have not been normalized or filtered.

Code Description

This algorithm works in the following manner: First, it obtains several images from the training set (figure 1). The code was been tested using bmp and jpg images. After loading these images, we find the mean face and perform several calculation (refer to code and tutorial). Next, we ask for the name of an image we want to recognize (note: the image must have the same size as the training images). The default path of your input image is your desktop. You could change it by modifying the code on line 159. We then project the input image into the eigenspace, and based on the difference from the eigenfaces we make a decision.
Please look at results of our algorithm and Adam's algorithm.

Final Words

Although our algorithm works, it is necessary to normalize the images and use some filtering techniques to obtain better results. We encourage you to modify our code to obtain better results. The following links will provide you with more examples and will give you a better idea about face recognition.

Useful links

Face Recognition by Man Kwok Ming and Cheung Chi Wai
Face Recognition Using Eigenfaces by Ilker Atalay
Real Time face Recognition using Eigenfaces by Raphael Cendrillon
Types of Face Recognition
History behind Eigenfaces
Face Detection
Carnegie Mellon Robotics Institute
MIT media laboratory
C++ libraries designed for computer vision research

Click here to email me