Project 5: Face Recognition Using Principal Component Analysis and Eigenfaces in MATLAB

Matteo De Angelis   
*College of Engineering*  
*University of Miami*Miami, USA  
mld163@miami.edu

Carter Falkenberg  
*College of Arts and Sciences  
University of Miami*  
Miami, USA  
ckf25@miami.edu

Kelly Rojas  
*College of Engineering*  
*University of Miami*Miami, USA  
knr20@miami.edu

*Abstract* — This project implements a face recognition system using the eigenfaces method based on Principal Component Analysis (PCA). The system is trained to detect and recognize facial images by projecting them into a reduced eigenspace and comparing them against known individuals. Two testing sets are used to evaluate the system: one containing known faces and another with unknown or non-face images. Key performance metrics include recognition accuracy, false positives, and the effect of varying the number of eigenfaces. The results demonstrate the effectiveness of PCA in dimensionality reduction and the trade-offs between recognition performance and computational efficiency. The project is implemented in MATLAB using grayscale face images, emphasizing theoretical understanding and practical application of eigenface-based recognition.

Keywords — Face Recognition, Principal Component Analysis (PCA), Eigenfaces, MATLAB, Image Classification, Dimensionality Reduction

# Introduction

Face recognition is critical in computer vision, with widespread applications in human-computer interaction, biometrics, and even security. Among the various techniques developed for face recognition, the eigenfaces method is highlighted due to its simplicity and effect8iveness in dimensionality reduction and feature extraction[1]. This project focuses on implementing the eigenfaces approach using PCA to develop a recognition system in MATLAB.

The system is trained on a dataset of facial images to compute the eigenfaces from PCA, and then tested on two sets of images: T1 (known faces), and T2 (unknown or non-faces). The primary objectives are to classify whether a test image is a face, and if it is a face, to identify it as a known individual from the training set. Only a subset of the most significant eigenfaces (those with the highest eigenvalues) is used, allowing analysis of recognition performance versus the number of retained components. Furthermore, the project investigates the system’s accuracy, failure cases, and sensitivity to the number of principal components. A simplified explanation of this approach is commonly found in educational resources that explain PCA’s role in facial classification[3].

# Theory

The eigenfaces methos is based on PCA, a statistical technique that transforms high-dimensional data into a lower-dimensional spaces while preserving the most important variance[2]. In the context of face recognition, each grayscale image is considered a high-dimensional vector, and PCA helps find a new set of orthogonal basis vectors (called eigenfaces) that best describe the dataset[1],[4].

First, the training images are mean-centered and then arranged into a matrix where each column represents a face vector. PCA is applied to this matrix to compute eigenvectors (eigenfaces) and their corresponding eigenvalues. The top eigenfaces (those with the largest eigenvalues) are retained, capturing the key variations among facial features[4].

A test image is projected into the reduced eigenspace to obtain a feature vector (its PCA coefficients). Recognition is then performed by comparing this vector to the feature vectors of training images using a distance metric like Euclidean distance. If the minimum distance is below a pre-defined threshold, the test image is classified as a known face. Otherwise, it is rejected as either an unknown or a non-face image. Additionally, PCA reduces noise and redundancy in facial data, and its comparison with other dimensionality reduction techniques further demonstrate its usefulness in features extraction[2],[3].

# Methodology

## Dataset Preparation

The dataset used for training and testing comprised of:

* 32 face images used for training
* 44 face images used for testing
* 7 non-face images used for testing

From each collection of face and non-face images that can be used for testing, only half of them respectively are chosen at random and used for testing. This entails that 22 face images and 4 non-face images are present in the testing dataset.

## Vectorization

All images are sized 640x480 pixels and are all vectorized to a 1D array. To try and achieve the best results the images used before vectorization are taken both in RGB color format and converted to a greyscale format. This leads with the following vector sizes depending if greyscale or color images are used:

* Color images:
* Greyscale images:

This is done consistently in two different processes.

## Mean Faces

To center the data before applying PCA, the mean face normalization vector is computed by averaging all training image vectors:

The training and testing datasets are then normalized by subtracting the mean:

## Principal Component Analysis

PCA is applied to center the data and to obtain the eigenvectors which are used to form the basis of the eigenface space. In order to retrieve only the most important eigenfaces to be used for testing, an explained cumulative variance of at least is retained. This is done by choosing the first values of that satisfy this formula:

where, is the minimum number of necessary to retain at least of the information, and is the total number of found using PCA.

The final eigenfaces found to represent the eigenface space are represented by:

## Face Recognition

Both the training and testing images are projected onto the eigenface space to obtain lower-dimensional feature representations.

To determine whether an image is a face, the reconstruction error is calculated by projecting the image back to the original space:

The error is defined as:

If the error exceeds a predefined threshold of 15000, the image is classified as “NON-FACE” image, if it is lower than that it is classified as “FACE” image.

For recognized “FACE” images, the Euclidean distance between the projected test vector and all projected training vectors is calculated:

The closest match is determined by finding the minimum distance, and the corresponding training image is considered the best match.

# Results

Results.

Text Box to add picture and description and keep it in the same column together

# Conclusion

Conclusion.

##### References

1. M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991. [Online]. Available: <https://www.face-rec.org/algorithms/PCA/jcn.pdf>
2. S. Raschka, "PCA vs LDA," *Python Machine Learning Book - GitHub FAQ*, 2016. [Online]. Available: <https://github.com/rasbt/python-machine-learning-book/blob/master/faq/lda-vs-pca.md>
3. A. Mishra, "Face Recognition using PCA (Eigenfaces)," *YouTube*, Sep. 18, 2021. [Online]. Available: <https://www.youtube.com/watch?v=g4Urfno4aTc>
4. Wikipedia contributors, "Eigenface," *Wikipedia, The Free Encyclopedia*, 2024. [Online]. Available: <https://en.wikipedia.org/wiki/Eigenface>