
Total # of points = 100.

Project description: Implement the Eigenface method you learned in class for face recognition. Use the training images in the face dataset provided to produce a set of Eigenfaces and coefficients, and then recognize the faces in the test data set using 1-NN (nearest neighbor) classifier. Here, we want to recognize the input face as one of the faces in the training set. To recognize the face in the input image, compute the Euclidean distances d_i between the Eigenface coefficients of the input image and the Eigenface coefficients of the training images. The input face is then recognized as the face in the training images with the smallest Euclidean distance. (You can skip the computation of \vec{I}_R and d_0 , and ignore thresholds T_0 and T_1 in the lecture slides.)

Face dataset: The images in the face dataset are grayscale images in *.jpg* format. Each pixel uses 8 bits for grayscale. The training set contains eight face images from eight different persons.

Programming language: You can use Python, C++/C, Java or Matlab to implement your program. If you plan to use another language, send me an email first. You cannot use library functions to perform any steps that you are required to implement. You can use library functions for the *reading*, *writing* and *displaying* of images, and for performing mathematical operations such as *matrix* and *vector* operations, and the computations of *eigenvalues* and *eigenvectors*.

You can work on the project by yourself or you can form a team of two. You can discuss with other classmates on how to do the project but every team is supposed to do their own coding and turn in their own project.

Submit on BrightSpace by the due date: If you work with a partner, only one of you needs to submit but put down both partners' names on the PDF document and the source code. Please submit the files separately on Brightspace. Do not put them into a ZIP file (or other compressed file formats.)

1. The source code file. Put comments in your source code to make it easier for someone else to read your program. Points will be taken off if you do not have comments.
2. Image files for the mean face m and the 8 Eigenfaces computed from the training images, in *.jpg* or *.bmp* formats.
3. A PDF file that contains
 - a) Instructions on how to run your program. If your program requires compilation, instructions on how to compile your program should also be provided.
 - b) Copy and paste the mean face and Eigenfaces in (2) above onto the PDF document.
 - c) The Eigenface coefficients of the training images $(\vec{\Omega}_i)$.
 - d) The Eigenface coefficients (Ω_i) and recognition result for each test image.
 - e) The recognition accuracy for the test dataset (# correct recognitions / # test images.)
 - f) Copy and paste your source code onto the PDF document.

(b) and (f) above are in addition to the source code file and output image files that you have to submit separately in (1) and (2) above.