Artificial Vision

Project 1 – Face Recognition

Goal:

To develop a face recognition system based on the techniques known as *eigenfaces* and *fisherfaces*.

Procedure:

1. Perform the acquisition of a face database from the students in the class (7 face images for each student). This set will be used in this project as the training and test sets. It may also be used a face database available in the Internet.
2. Proceed with the normalization of the face images from the training set (rotation, scaling, selection) so that the faces comply with the format defined by MPEG-7 recommendation. That is, each face is represented by a monochrome image (256 levels) with 56 rows and 46 columns, with both eyes, right and left, perfectly aligned horizontally and located in line 24, columns 16 and 31, respectively.
3. Compute the mean face vector μ (column vector with size of 1×2576) and the transformation matrix W_{pca} (with size of 2576×*m*), containing the first *m eigenfaces* (where *m* can be a number between 10 and 20). Use the procedures presented in the slides of the course. View the *m*+1 obtained vectors by interpreting them as face images (*eigenface*) and prove that they form an orthogonal basis.
4. Preform the projection of some of the training set faces on the face subspace and observe their reconstructions. Compute (and view) the error face (difference between the original face and its reconstruction). Show that the error face is orthogonal to the face subspace.

5. Develop a Nearest-Neighbor (NN) classifier based on the feature vectors resulting from the *m* projection coefficients. Consider an arbitrary number of classes, *c* (involving all students in the class or just a part of them).

6. Implement a function with the following interface:

   list = function faceEngine*(faceq, N)*

   This function has 2 input parameters: a normalized face image, *faceq*, and an integer, *N*. It returns a column vector, *list*, containing *N* indices pointing to faces from the database, sorted in descending order of "similarity". See the results

7. Repeat the previous points, but now using the algorithm called *fisherfaces*. Use the procedures presented in the slides of the course

8. Test the implemented classifiers using the test set (different from the training set).