

MVA 2008 - 2009

Object recognition and computer vision

Assignment 3: Eigenfaces for recognition

Jean Ponce, Cordelia Schmid and Josef Sivic

Due date: December 9th 2008

(adapted from Svetlana Lazebnik, UNC)



The goal of the assignment is to implement a nearest-neighbor face recognition algorithm based on Eigenfaces as discussed in lectures. We will be using a database consisting of 10 images each of 40 different people. The image resolution is 32x32, corresponding to the extrinsic dimension of 1024. The task is to correctly identify the person from a test image using a low-dimensional eigenface representation.

Homework materials

Here are the [materials](#), including the face database ([from here](#)), training and test indices, and sample code.

Directions

1. Load the face database and partition it into training and test sets (this is already done in the sample code provided for the assignment). We will be using 7 images per person for training and 3 images for testing.
2. Find the mean face image. Perform PCA on the training faces and extract top K components. You can use the provided MATLAB `princomp` function.
3. Compute its K-dimensional projection of the test images onto the face space.
4. For each test image, find the training image that is "closest" (in the sense of Euclidean distance) to the test image in the face space, and assign the label (person index) of the training image to the test image.
5. As a baseline, implement nearest neighbour classifier based on normalized cross-correlation between the original images. You can use MATLAB function `corr2` to compute cross-correlation between images.

What to hand in

- The picture of the mean face and top 20 eigenfaces computed by PCA.
- A plot of the nearest-neighbor classification rate as a function of K . You can choose any sampling of values K from 1 to 1024, as long as it captures the trend of how classification performance changes as a function of K (i.e., we expect performance to be poor for extremely low K , but then to rise very rapidly and level off at some point). Include the classification rate of the cross-correlation baseline in the plot. You can use the MATLAB function `semilogx` instead of `plot` to draw the x-axis (K) in the logarithmic scale.
- Pictures of incorrectly classified faces and their nearest neighbors from the training set for $K=100$. The sample code can be easily adapted to do this.
- As usual, include your MATLAB code and a brief report addressing the above issues.