Digital Image Processing
CS390S

Feng Jiang

2018 Spring

METROPOLITAN STATE UNIVERSITY℠
OF DENVER

Review Face recognition

- Paper 0
- Paper 1

PCA Face Recognition
Read the two papers!

***Given*:**

M training images from **m** different persons with same size: row-by-col.
Mp = desired number of principal components, usually **Mp=m**

Feature Extraction Steps

1. For each training image, change to 1 column vector, each vector will have row*col length
2. For each person, calculate the average vector Xi
3. Calculate the mean vector of all the persons in the system Me=(X1+X2+…+Xm)/m
4. Let Ai=Xi-Me
5. Calculate the eigen vectors of A*A' store it as P
6. Calculate the weight of the training data projected into eigenspace wt_A = P'*A

Feature Match Steps

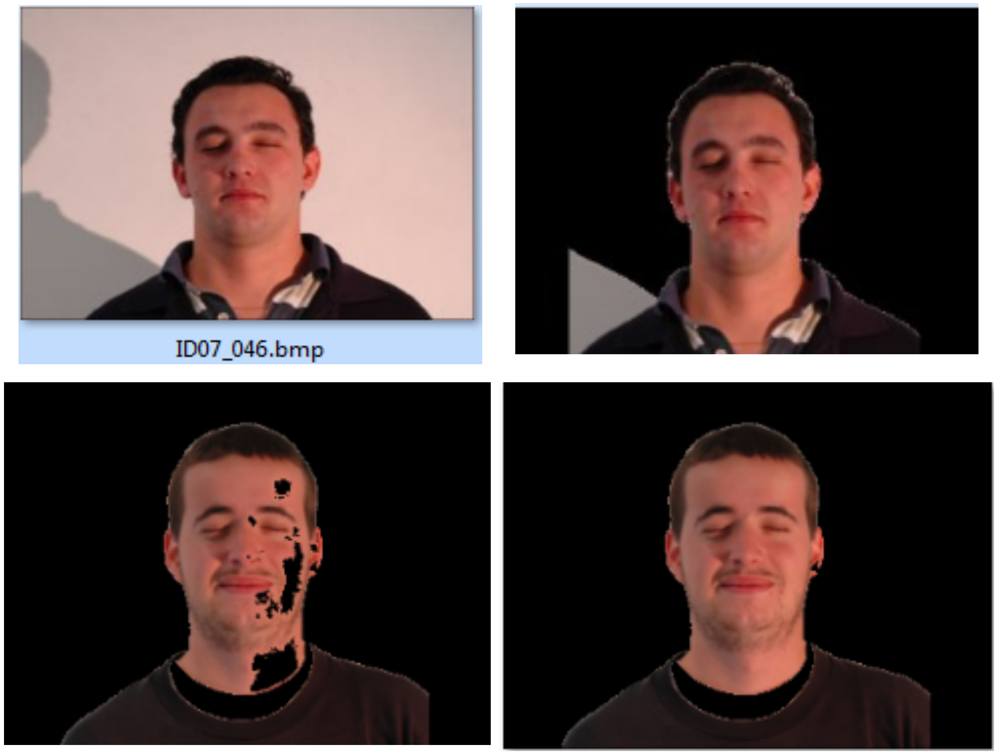
1. For the input image Im, change to 1 column vector: Y
2. Calculate B = Y-Me
3. Calculate weight of the input data projected into eigenspace wt_B=P'*B;
4. Calculate the euclidean distance for the input image: eud(i) = sqrt(sum((wt_B-wt_A(:,i)).^2));
5. Find the smallest euclidean distance j, the input face is from person j.

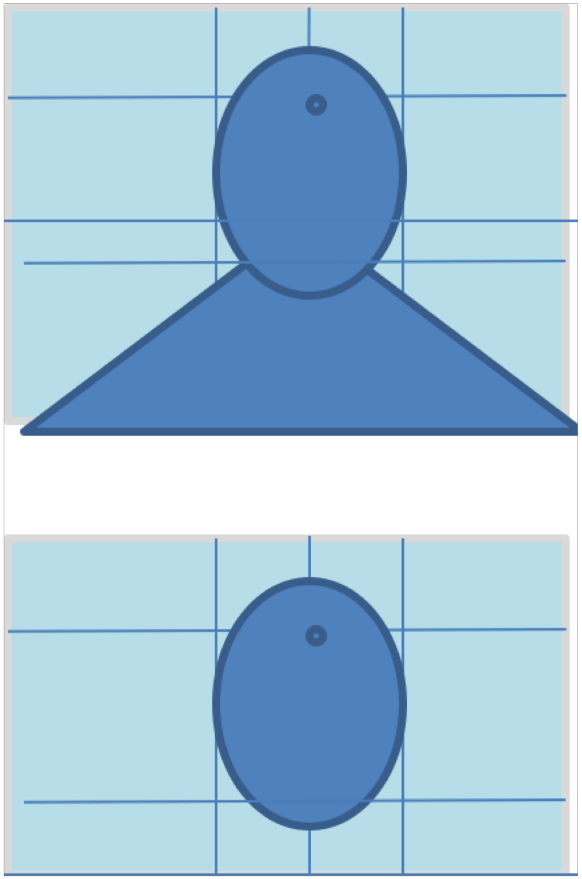
PCA Face Recognition

- Read the two papers given
- Explain the steps of PCA FR in slide

PCA Face Recognition

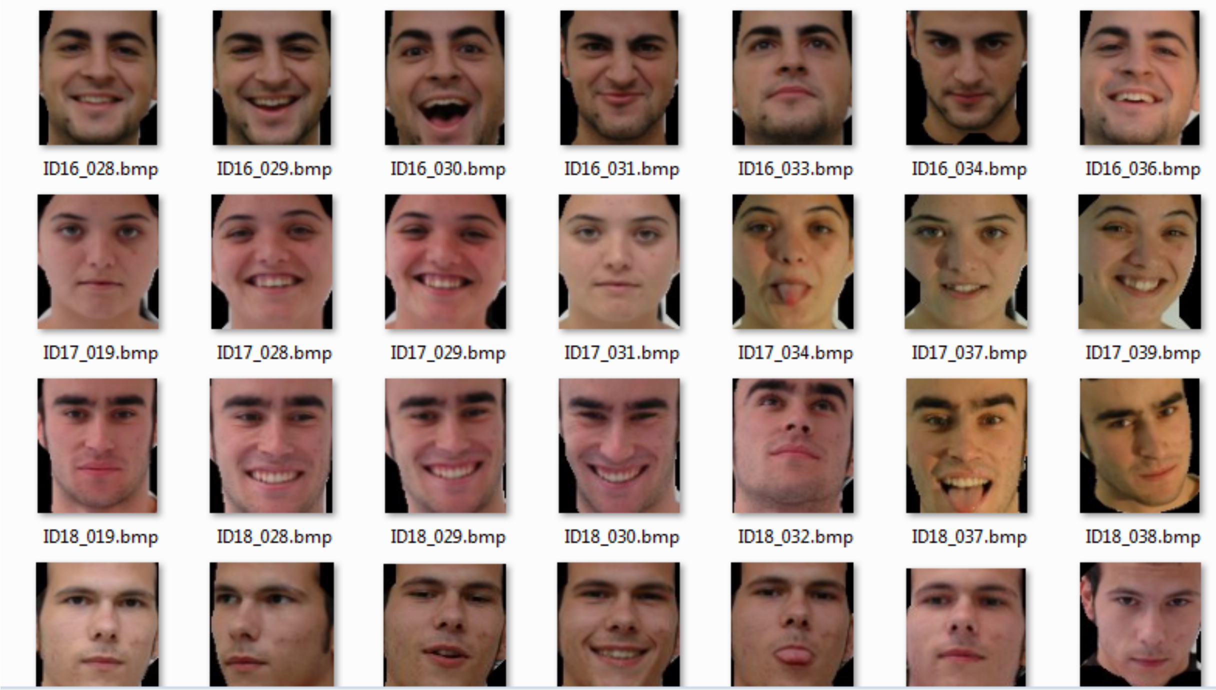
- Face detection





PCA Face Recognition

- Face detection

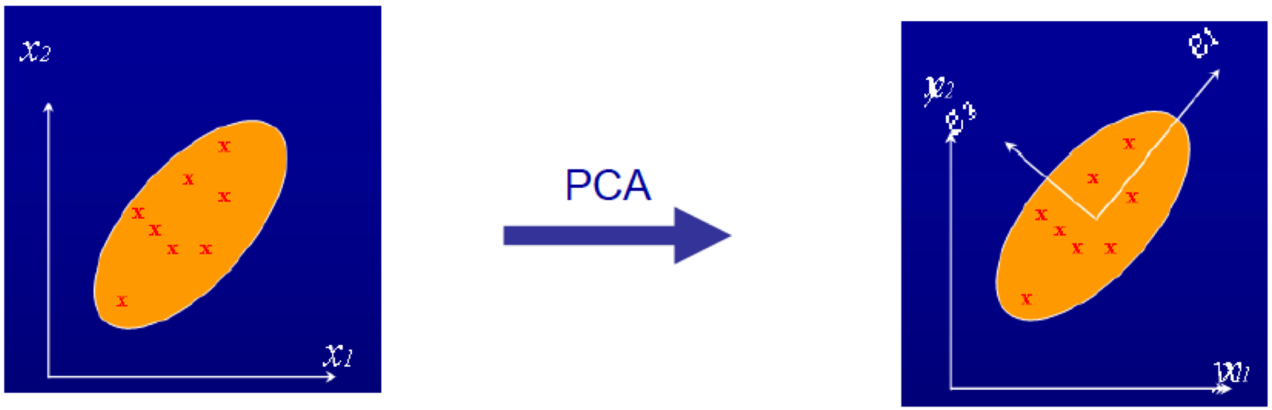


PCA Face Recognition

- Principal component analysis
  - Eigen Face/Hotteling Transform/Karhunen-Loeve Method

- Principle: find an orthogonal coordinate system such that data is approximated best and the correlation between different axis is minimized.

PCA Face Recognition



- Define a new origin as the mean of the data set
- Find the direction of maximum variance in the samples (e1) and align it with the first axis (y1),
- Continue this process with orthogonal directions of decreasing variance, aligning each with the next axis

- Thus, we have a rotation which minimizes the correlation.

Paper 0

- What’s the principle component?
- How to match a new face?
- How to be more efficient in face features representation (coding…) ?

Paper 0

another. The approach transforms face images into a small set of characteristic feature images, called "eigenfaces", which are the principal components of the initial training set of face images. Recognition is performed by projecting a new image into the subspace spanned by the eigenfaces ("face space") and then classifying the face by comparing its position in face space with the positions of known individuals.

Paper 0

In the language of information theory, we want to extract the relevant information in a face image, encode it as efficiently as possible, and compare one face encoding with a database of models encoded similarly. A simple approach to extracting the information contained in an image of a face is to somehow capture the variation in a collection of face images, independent of any judgement of features, and use this information to encode and compare individual face images.

In mathematical terms, we wish to find the principal components of the distribution of faces, or the eigenvectors of the covariance matrix of the set of face images. These eigenvectors can be thought of as a set of features which together characterize the variation between face images. Each image location contributes more or less to each eigenvector, so that we can display the eigenvector as a sort of ghostly face which we call an *eigenface*. Some of these faces are shown in Figure 2.

Eigenvector, Eigen Value

- A matrix, v column vector, lambda Eigen value for v
- Geometrically an eigenvector, corresponding to a real nonzero eigenvalue, points in a direction that is stretched by the transformation and the eigenvalue is the factor by which it is stretched.

$$A\mathbf{v} = \lambda\mathbf{v}.$$

- (AB)’=B’A’

Eigenface

Eigen Face Image

PCA Face Recognition

Given:

M training images from **m** different persons with same size: row-by-col.
Mp = desired number of principal components, usually **Mp=m**

Feature Extraction Steps

1. For each training image, change to 1 column vector, each vector will have row*col length
2. For each person, calculate the average vector Xi
3. Calculate the mean vector of all the persons in the system Me=(X1+X2+…+Xm)/m
4. Let Ai=Xi-Me
5. Calculate the eigen vectors of A*A' store it as P
6. Calculate the weight of the training data projected into eigenspace wt_A = P'*A

Feature Match Steps

1. For the input image Im, change to 1 column vector: Y
2. Calculate B = Y-Me
3. Calculate weight of the input data projected into eigenspace wt_B=P'*B;
4. Calculate the euclidean distance for the input image: eud(i) = sqrt(sum((wt_B-wt_A(:,i)).^2));
5. Find the smallest euclidean distance j, the input face is from person j.

PCA Face Recognition

(version 2--Recommended)

***Given*:**

M training images from **m** different persons with same size: row-by-col.
Mp = desired number of principal components, usually **Mp=m**

Feature Extraction Steps

1. For each training image, change to 1 column vector, each vector will have row*col length
2. For each person, calculate the average vector Xi
3. Calculate the mean vector of all the persons in the system Me=(X1+X2+…+Xm)/m
4. Let Ai=Xi-Me
5. Calculate the eigen vectors of A' *A store it as P2
6. Calculate the weight of the training data projected into eigen space wt_A = P2*(A'*A)

[Step 6 equals to following 2 sub steps:
Calculate the eigen vectors of A*A' as: P = A*P2 and
Calculate the weight of the training data projected into eigenspace wt_A = P'*A]

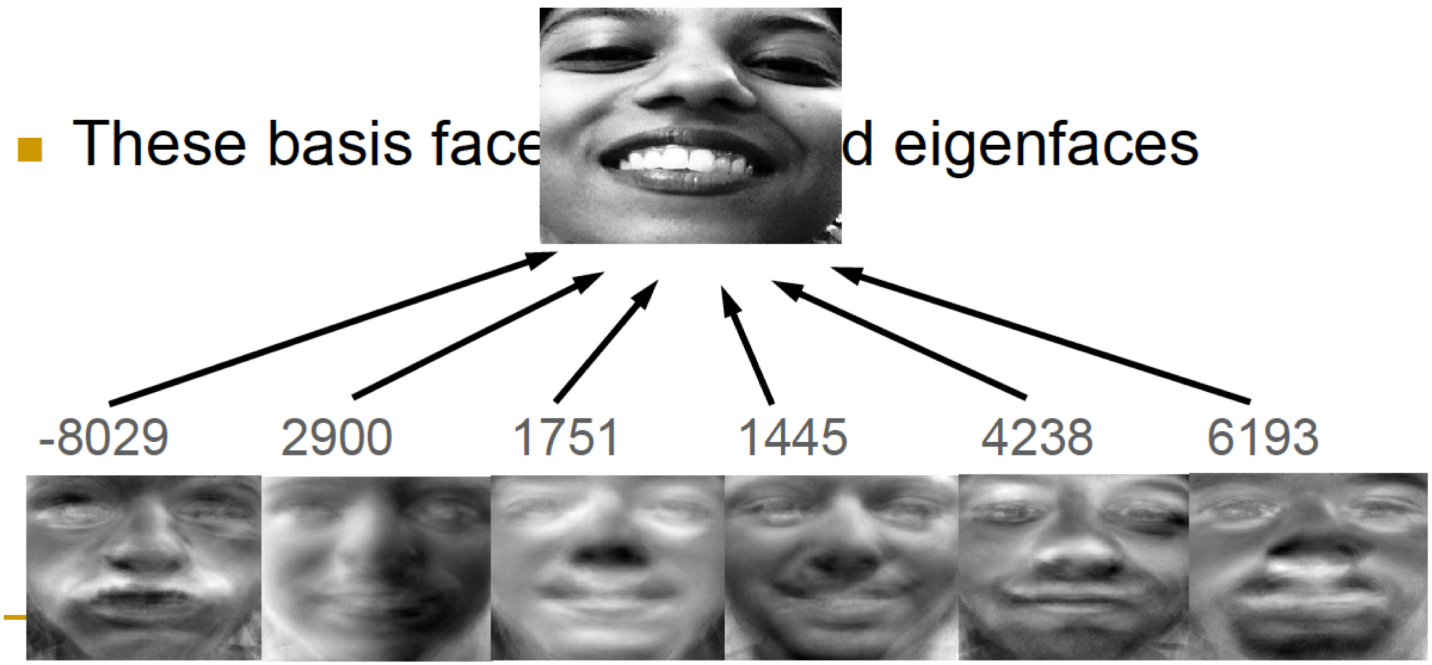
(Note: In this approach, if you would like to show the eigen faces, you can calculate the eigen faces as: P = A*P2)

Feature Match Steps

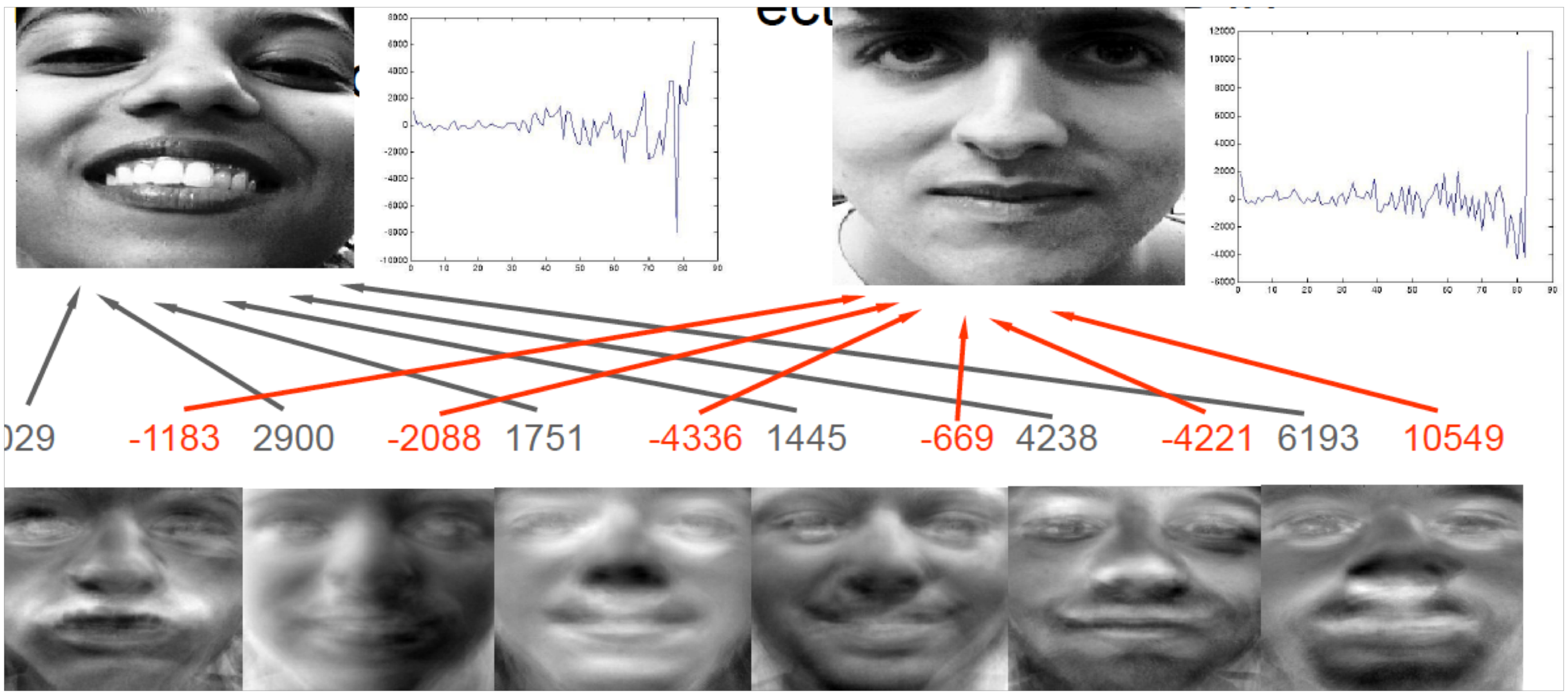
1. For the input image Im, change to 1 column vector: Y
2. Calculate B = Y-Me
3. Calculate weight of the input data projected into eigenspace wt_B=P'*B;
4. Calculate the euclidean distance for the input image: eud(i) = sqrt(sum((wt_B-wt_A(:,i)).^2));
5. Find the smallest euclidean distance j, the input face is from person j.

PCA Face Recognition

- Think of a face as being a weighted combination of some “component” or “basis” faces
- These basis face d eigenfaces



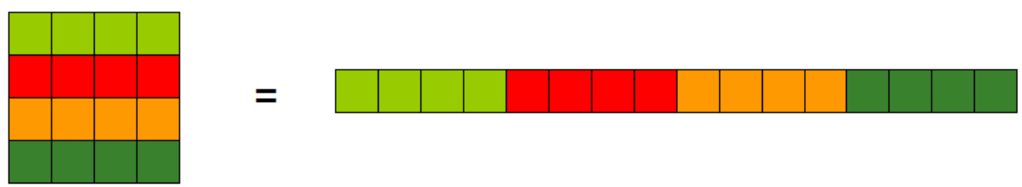
PCA Face Recognition



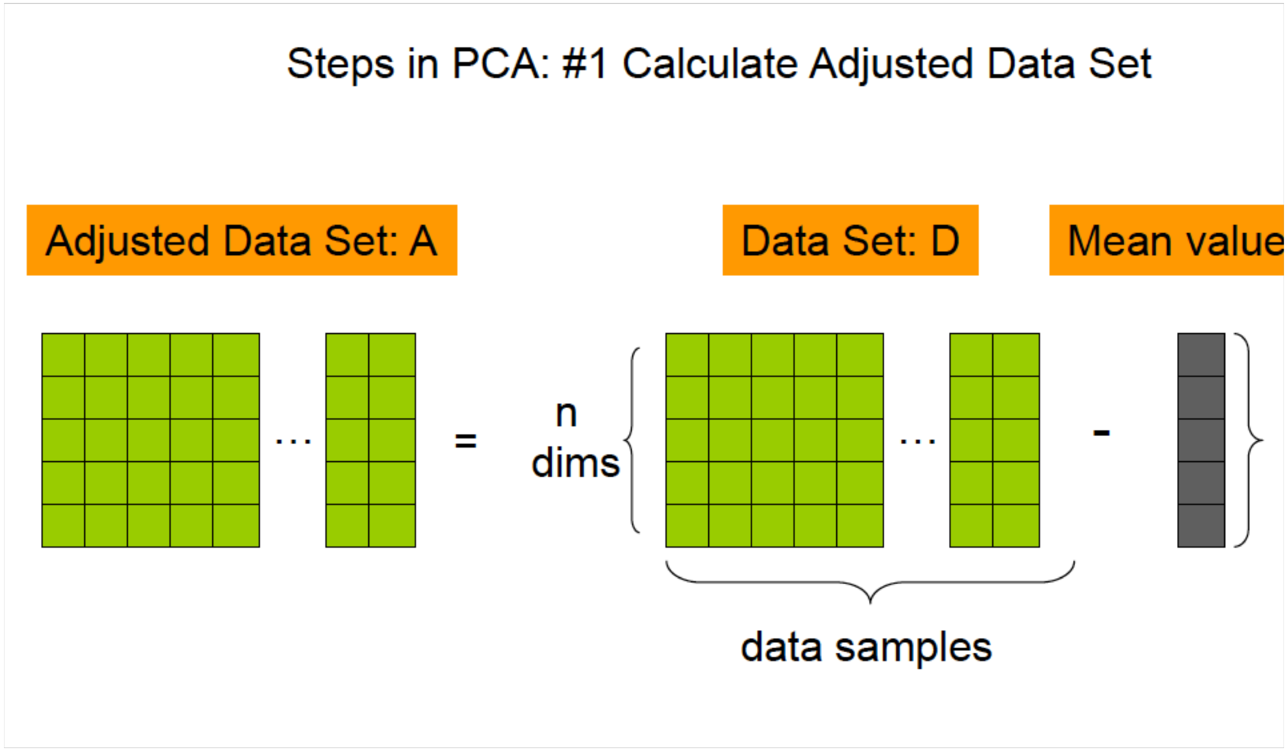
PCA Face Recognition

Matrix to vector

"Vectorizing" a matrix

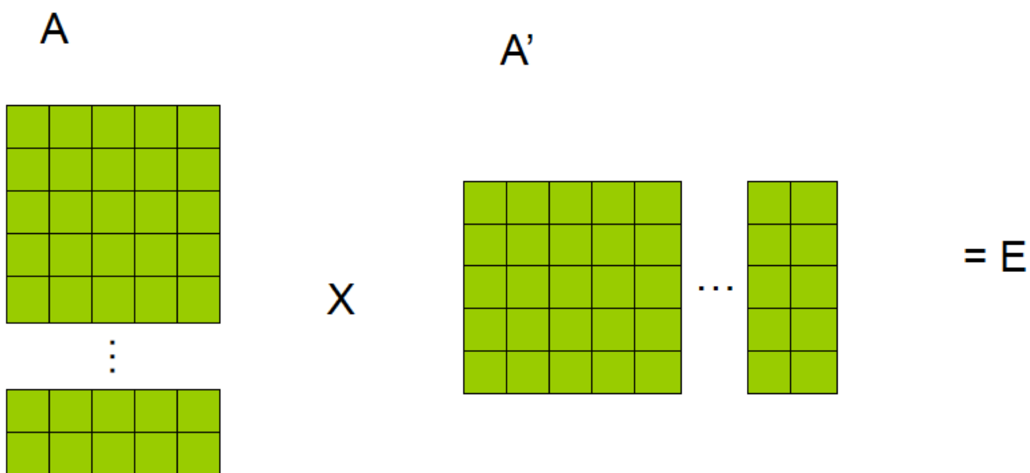


PCA Face Recognition



PCA Face Recognition

Dimension of A*A': N*N



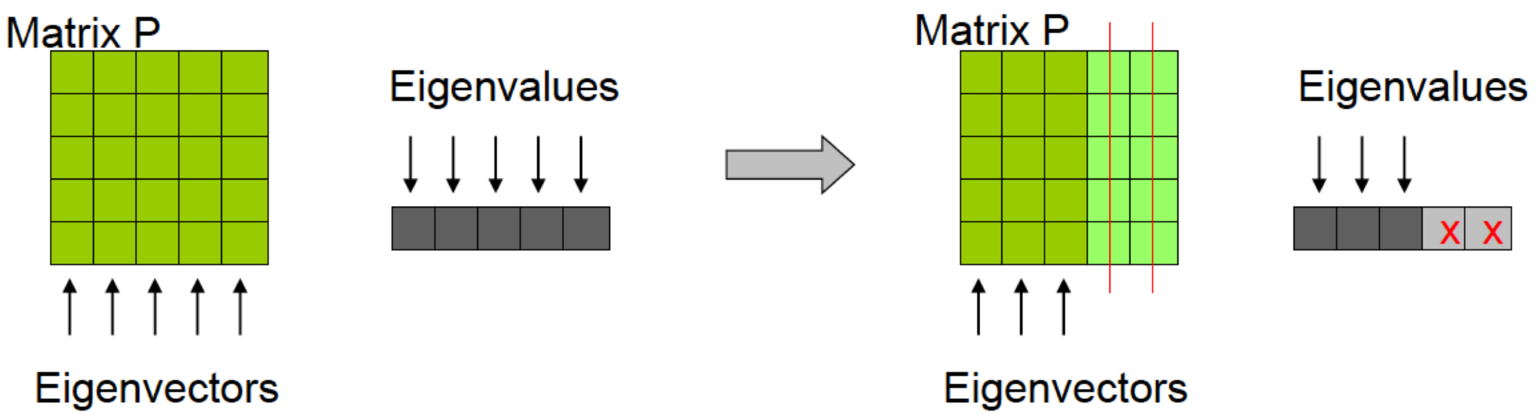
Dimension: M*N

Dimension: N*M

NOTE: N: number of dimensions/image; M: number of samples
Usually: M<<N

PCA Face Recognition

Using SVD find the eigenvectors of Matrix E



If some eigenvalues are 0 or very small, we can essentially discard those eigenvalues and the corresponding eigenvectors, hence reducing the dimensionality of the new basis.

PCA Face Recognition

Steps in PCA: #4 Transforming data set to the new basis

$$F = P^T A$$

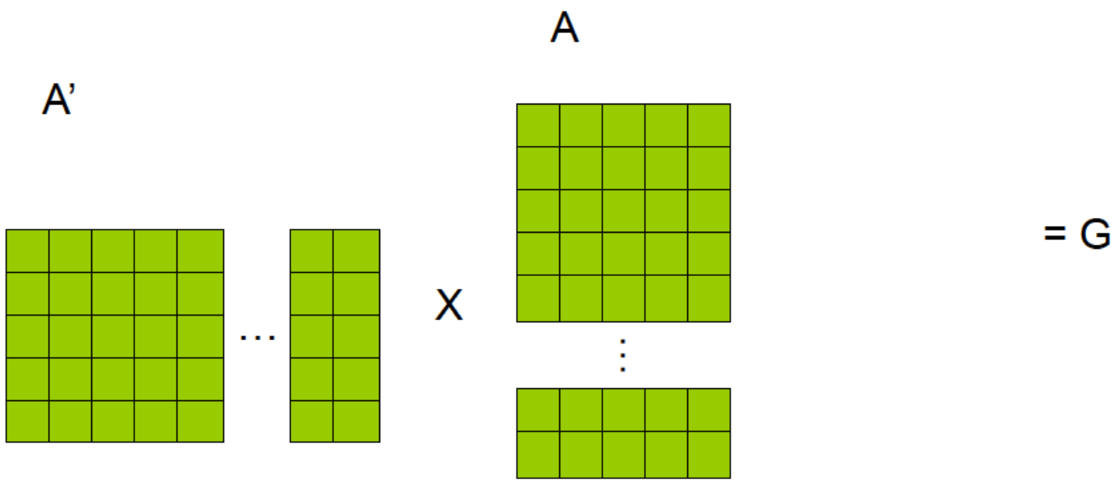
where:

- F is the transformed data set
- P^T is the transpose of the P matrix containing the eigenvectors
- A is the adjusted data set

Note that the dimensions of the new dataset, F, are less than the data set A

PCA Face Recognition

Dimension of A'*A: M*M. Since M<<N, so G is much smaller than E

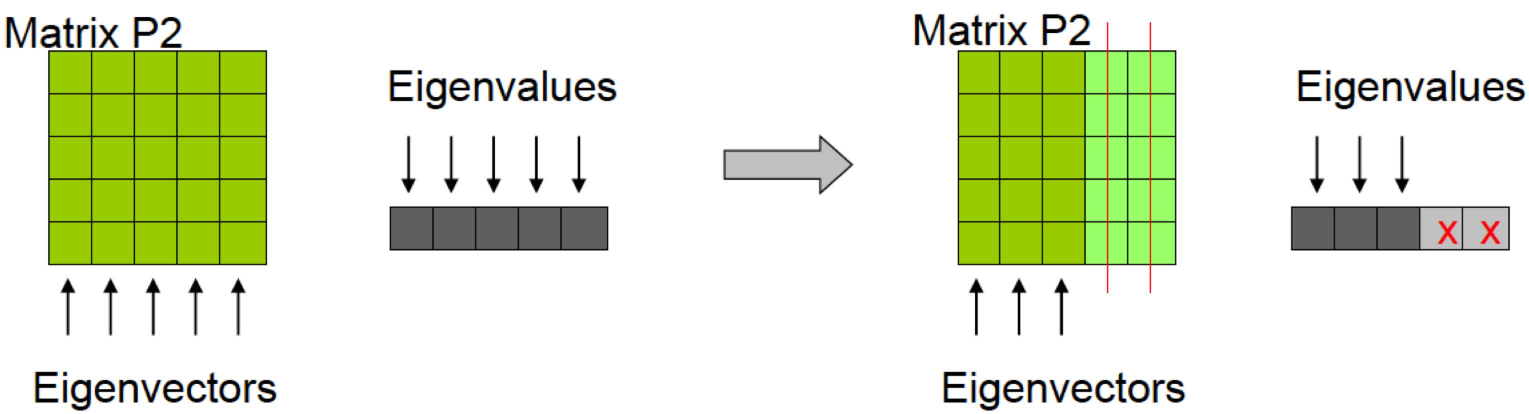


Dimension: M*N   Dimension: N*M

NOTE: N: number of dimensions/image; M: number of samples

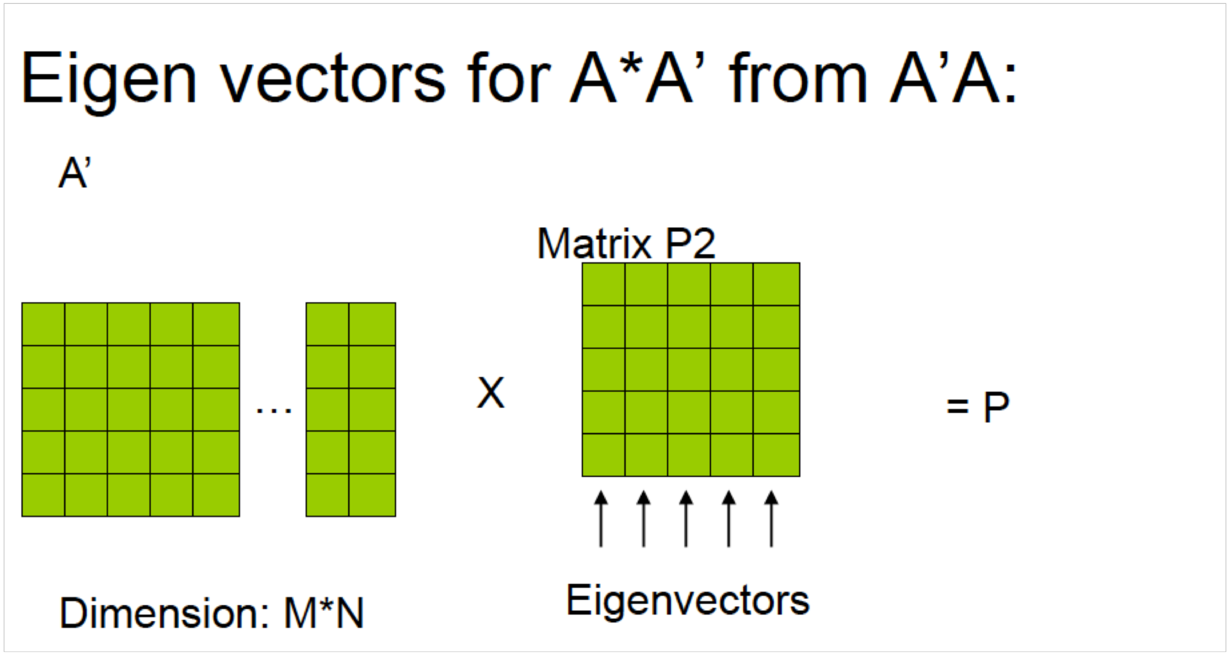
PCA Face Recognition

Using SVD find the eigenvectors of Matrix G



If some eigenvalues are 0 or very small, we can essentially discard those eigenvalues and the corresponding eigenvectors, hence reducing the dimensionality of the new basis.

PCA Face Recognition



PCA Face Recognition

Steps in PCA: #4 Transforming data set to the new basis

$$F = P^T A'$$

where:

- F is the transformed data set
- P^T is the transpose of the P matrix containing the eigenvectors
- A is the adjusted data set

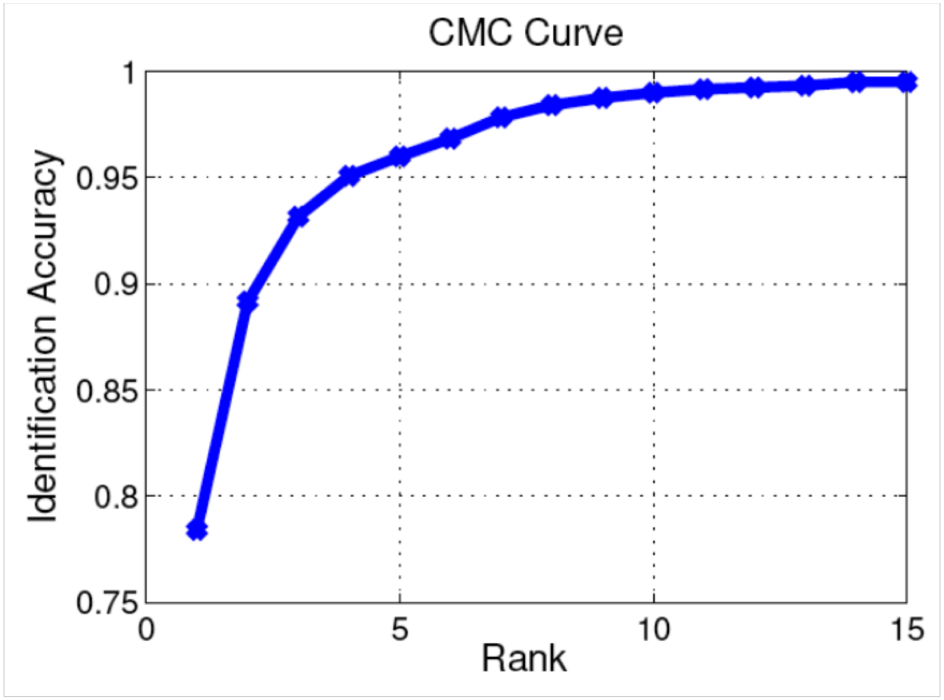
Note that the dimensions of the new dataset, F, are less than the data set A

CMC curve

- Performance of a closed-set identification system is summarized using Cumulative Match Characteristic (CMC) curve

- The resulting scores are sorted and ranked
- Determine the rank at which a true match occurs
- True Positive Identification Rate (TPIR): Probability of observing the correct identity within the top K ranks
- CMC Curve: Plots TPIR against ranks
- CMC Curve: Rank-based metric

CMC curve


CMC Curve
Identification Accuracy
Rank
1
0.95
0.9
0.85
0.8
0.75
0
5
10
15

CMC curve

```
for i=1:10  %% if show CMC of the 1st to 10th rank matching number
cmc(i)=sum(match(1:i))/num_imt;
end
```

CMC curve

```
Index of person ------- ith person
for ii=1:num_p   %% weight compare wtb and wta(i)
 eud(ii)=sqrt(sum((wtb-wta(:,ii)).^2));
end
[cdata index(i)]=min(eud);  %% find minimum eud's index
 %%% right result by observation is 1 1 2 3 4 %%%%%
result=[1 1 2 3 4];
%%%%%%%%%%%%%%% CMC calculation %%%%%%%
   if index(i)==result(i)
       match(1)=match(1)+1;%%%%%%%first rank matching number
   else
       [svals,idx]=sort(eud(:));
       index2(i)=idx(2);
       if index2(i)==result(i)
           match(2)=match(2)+1;%%%%%%%second rank matching number
       end
   end
end
```

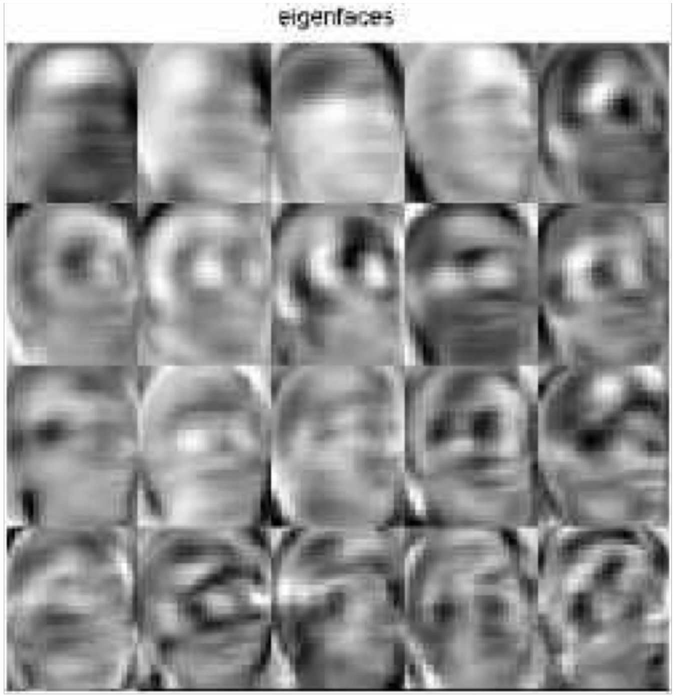
Project 1

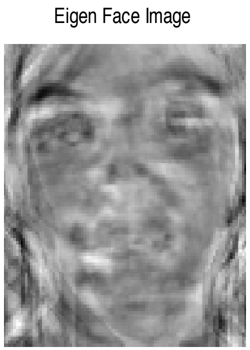
- Training:
  - Select 20 or above persons in large dataset from enroll set
  - (See imagewith44persons)
  - Include your own face as one person sample
  - Use 5 images for each person to train the system

- Testing：
  - Select at least 10 persons from testing dataset and at least 1 image from each persons

Project 1

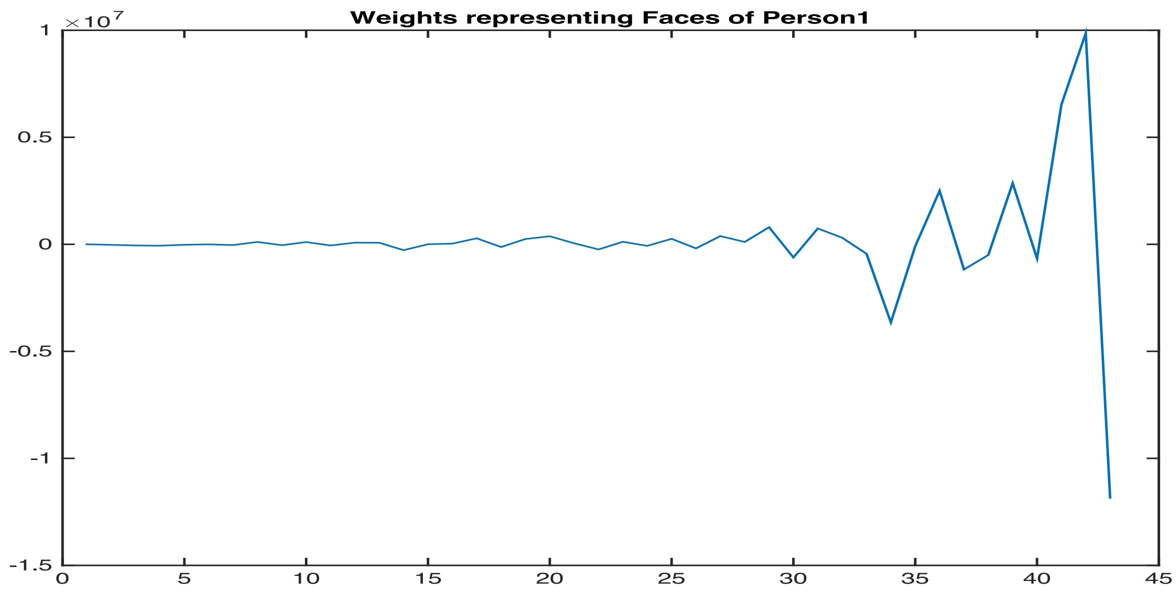
- Show the Eigenfaces generated by your training process





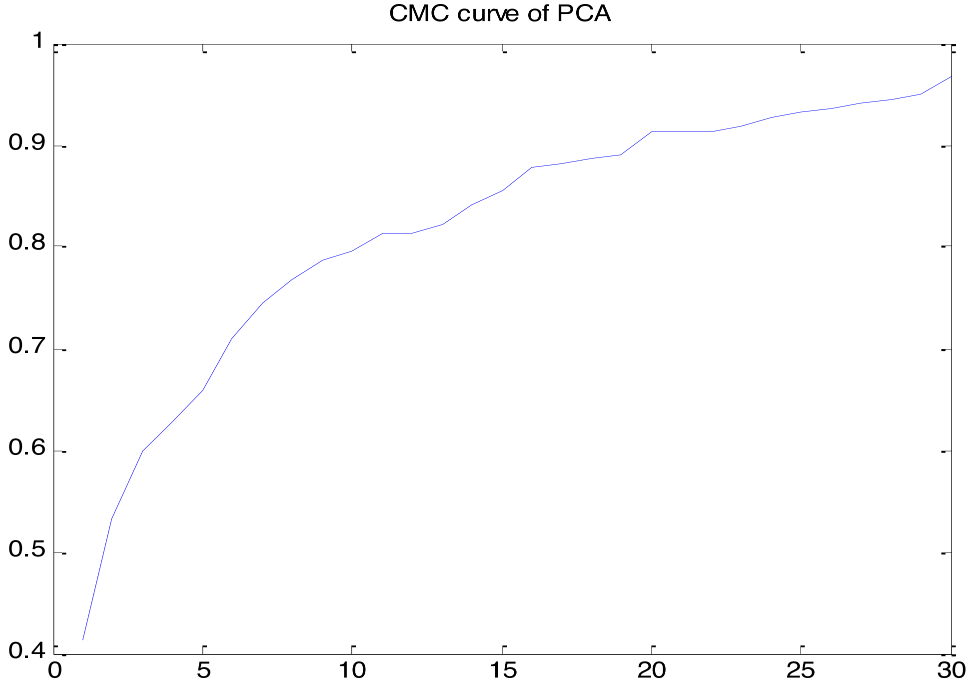
Project 1

- Plot the weights representing person 1 using eigen faces



Project 1

- Plot the CMC curve for your testing result
- Provide at least rank 1 to 10 matching results



Project 1 **Due Oct 27th**

- Report (include plot, figure, analysis, discussion of result) 10/25
- Performance (able to run the program, correctness of algorithm) 10/25

- Presentation (**Due Oct 22nd 11:59pm**) 5/25
  - Name on slide! peer review score (1-5)
  - 3 minutes
  - Design + **special approaches** + performance analysis and applications

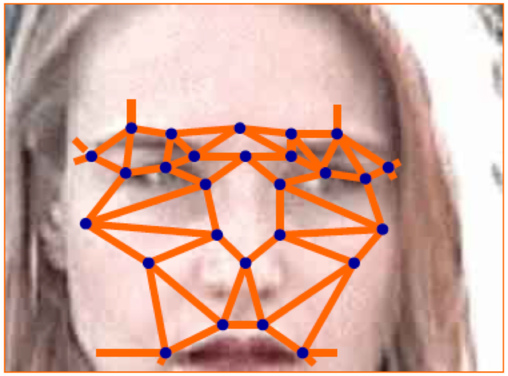
Face Recognition

- Holistic matching methods
- Feature-based (structural) matching methods
- Hybrid methods
- Other methods

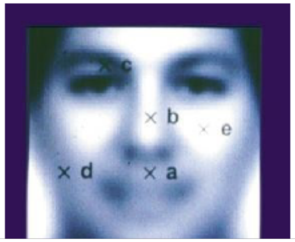
Face Recognition

Feature-based (structural) matching methods

- The Structure of the face

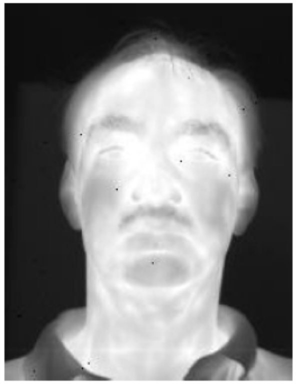


- The skin of the face

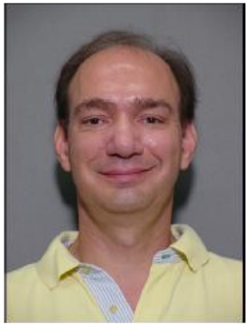


Face Recognition

- Infrared Face



Infrared imagery: Immune to lighting



Easy