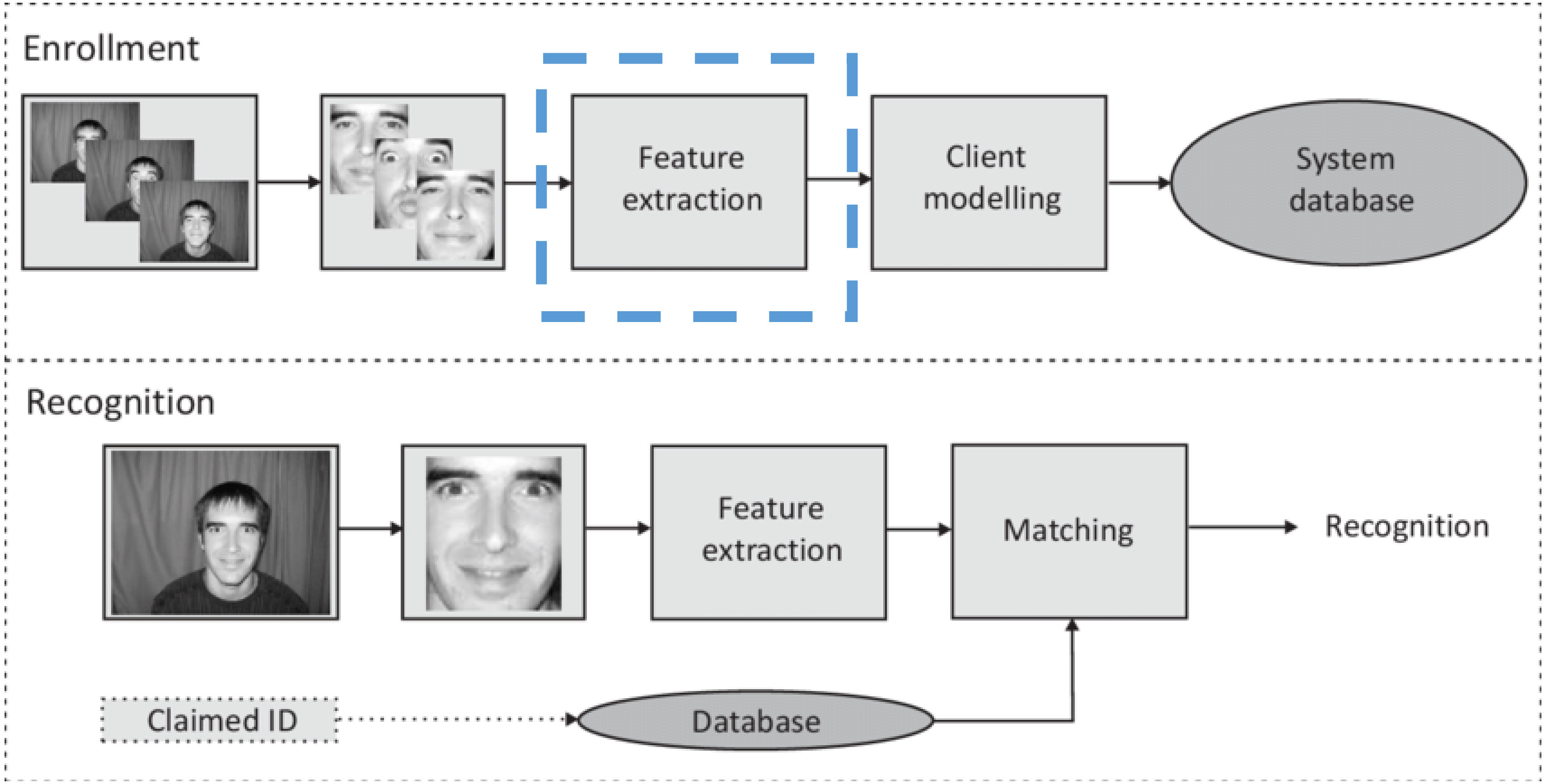
2D Face Recognition

Part 1

Ravitha N

Components of a Typical Biometric System



What we get and What we don't get??

Advantages

- Low cost
- Contactless
- Acceptability

Disadvantages

- Not highly secure
- Face changes over time for voluntary (or) non voluntary reasons

Approaches

Global Recognition

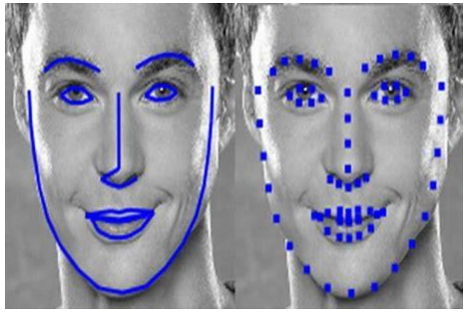
- PCA
- LDA

Process the image as a whole

Local Recognition

- Geometric techniques
- Elastic graph matching techniques

Divide image into patches and process them separately to extract features



Hybrid Techniques

Global Recognition Approach

Use Raw Pixel Intensity Values

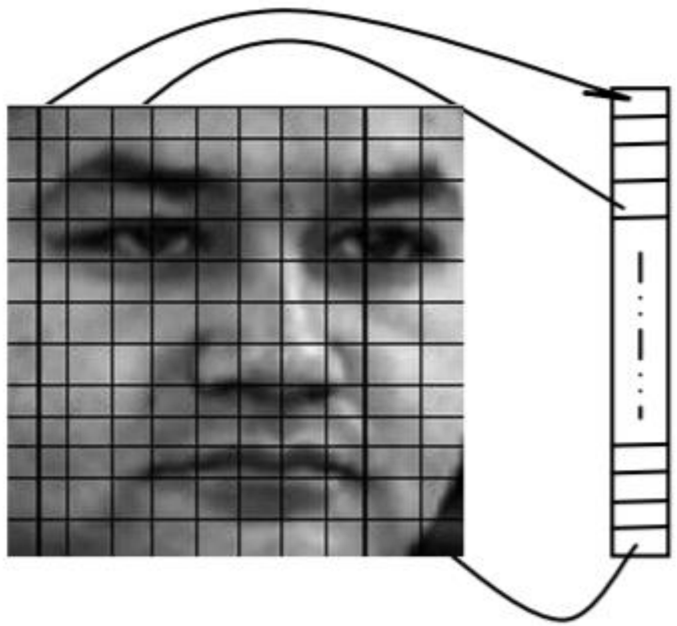
Convert the image into single dimensional global vector and
use the vector for recognition

PCA (Principal Component Analysis is used for dimensionality reduction

Dimension refers to the size of the vector
If an image is 4*4, size of the vector would be 16
Typical face image would be of size 256 * 256 , then global vector
would contain 65536 pixels

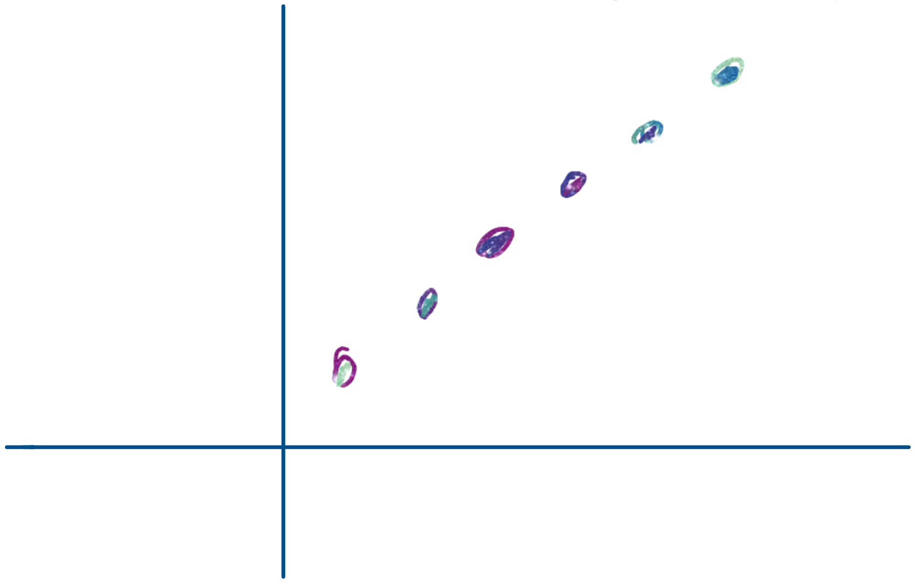
Why to reduce dimensions?

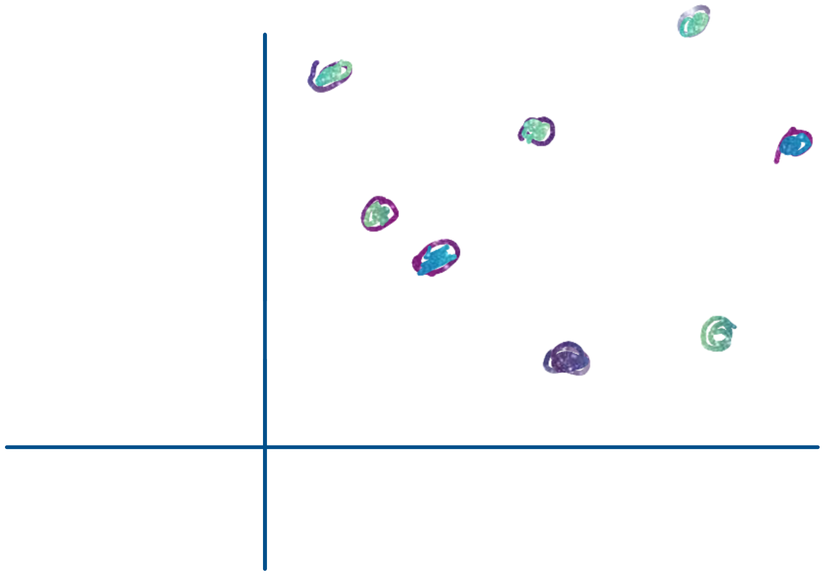
1. Decorrelate Data
2. Reduce the space dimensions



Decorrelating data

- Decorrelating Features
- Features that provide redundant information to be removed (or) it would affect the recognition process.





Dimensionality Reduction

- Represent Data (images) with less number of features
- Find alternate representation for Data

  Original Representation {B1,B2,B3,B4}

  (i.e,) Every data point represented by four features

  Need for alternate representation (reduce features)

  PCA finds alternate representation {D1,D2,D3,D4}with same number of features as that of original representation. Features are reduced by ignoring the feature vector which carries less variance.

Dimensionality Reduction

- Note:
  - Alternate representation are linear combination of original representation
  - Basis of alternate representation is always orthogonal in nature
  - Alternate representation in PCA will be of same size as that of original representation

Pause Moment

- Self Evaluate

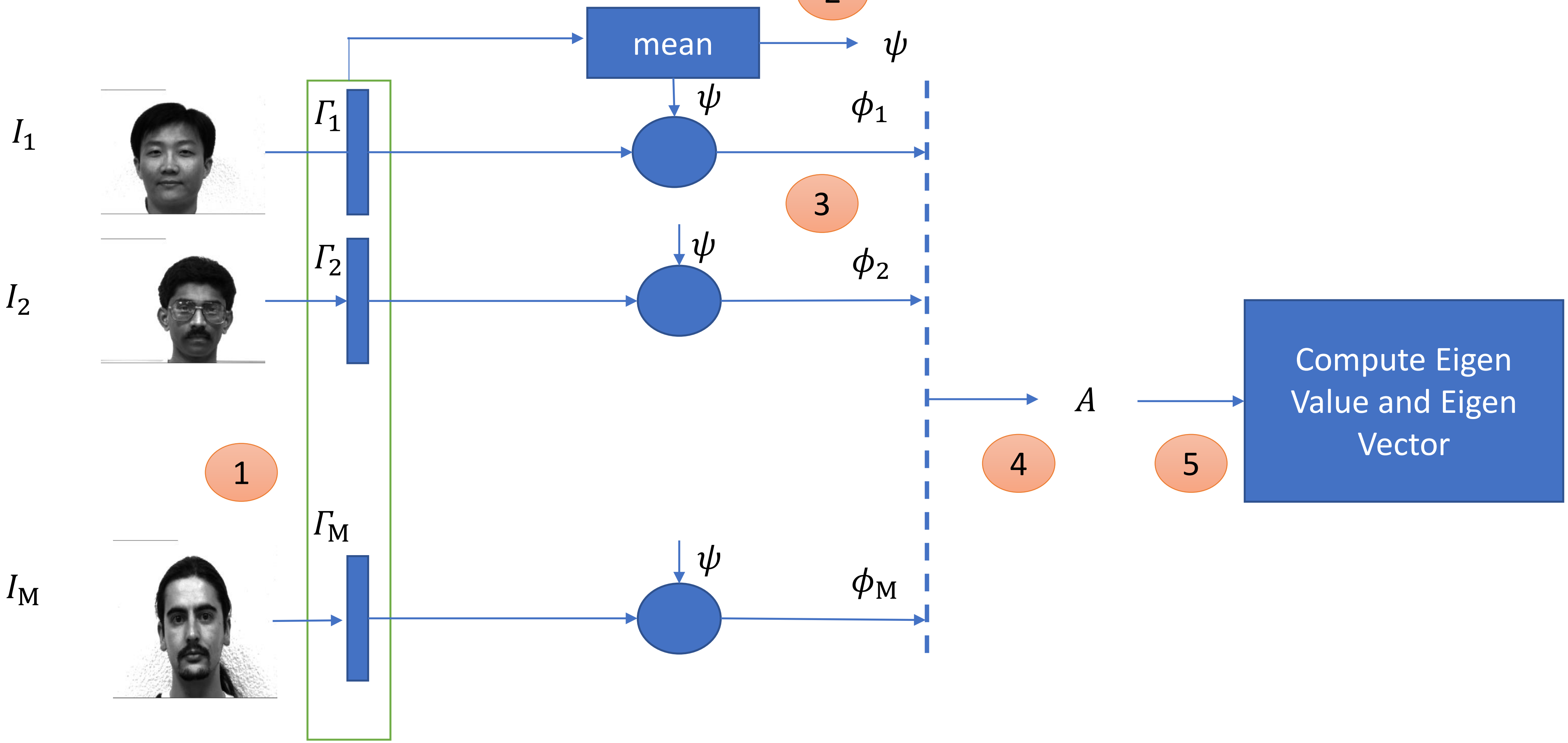
1. If an image is of size 4*4, how many features are there in global approach?
2. If PCA is applied for dataset which contains 100 features , How many eigen vectors it will provide? What would be the size of the eigen vector?

Answers

1. 16 (each pixel is considered as a feature)
2. It will provide 100 eigen vectors, each is a linear combination of 100 features. So, size will also be 100

Face Recognition Using PCA – Learning Phase

- Extract Principal Component from learning set (eigen faces)
- Projection of face onto principal component allo[illegible]xtraction of features.



- Learning phase (Consider Learning set has M images each of size N*N)
    1. Convert Images I_i in the learning set as columnar vector Γ_{i}
    2. Compute Mean Vector

$$\psi = \frac{1}{M}\sum_{i=1}^{M}\Gamma_i$$

    3. Find Mean subtracted images

$$\phi_i = \Gamma_i - \psi$$

    4. Construct the observation matrix

$$\underset{N^2 \times M}{A} = [\phi_1, \phi_2 \cdot \cdots \phi_M]$$

5. Compute Eigen value and Eigen vector

Decomposition of Covariance Matrix

$$C = AA^T$$

Finding AA^T is complex

If M<< N^2 then only M-1 eigen vectors would be significant

Solution Proposed

Compute eigen vector V for $A^T A$

Find eigen vector of AA^T as

$$u_k = Av_k | k = 1,2 \cdots M - 1$$

Pause Moment

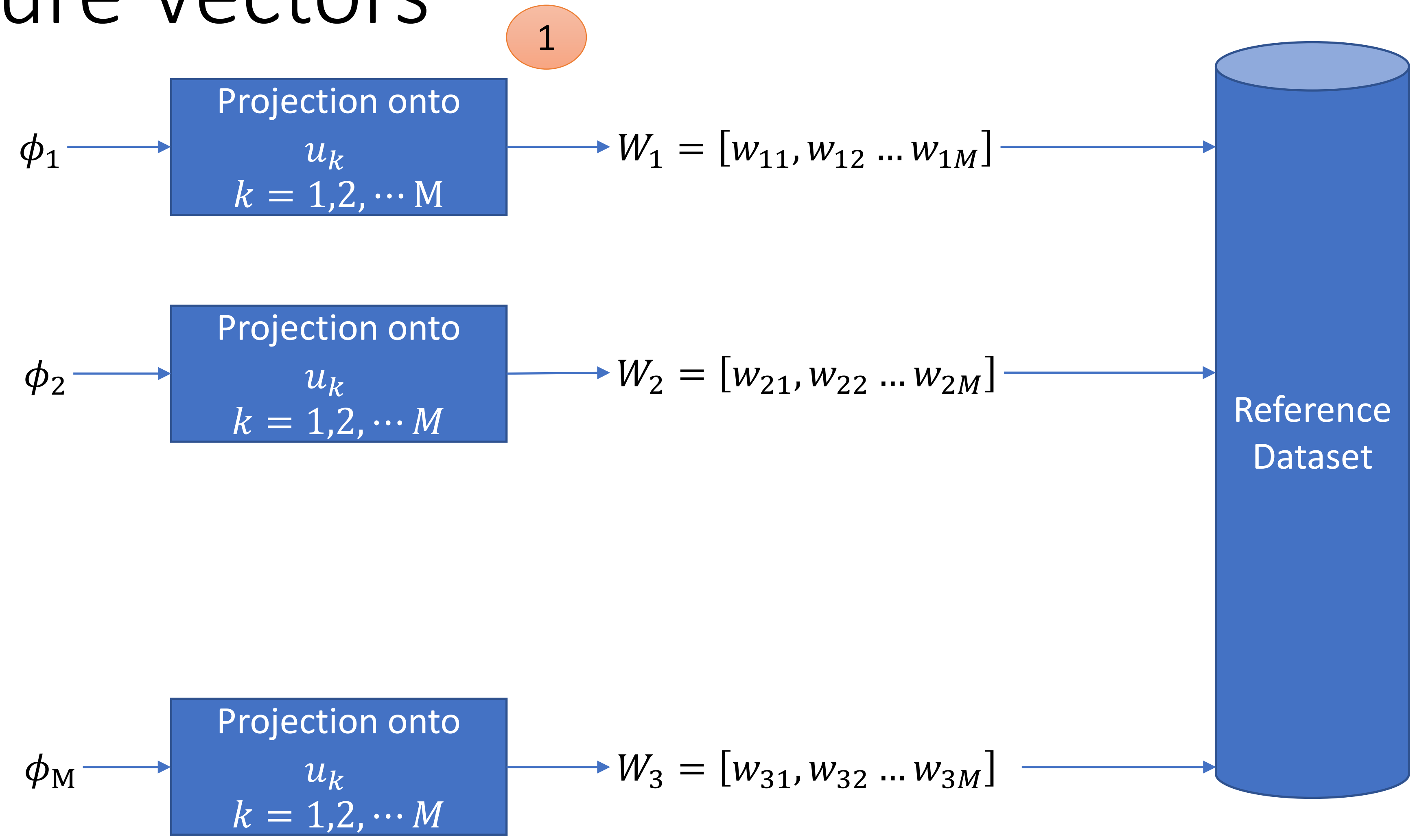
- Self Evaluate

1. According to the eigen faces approach, if the dataset contains 30 images of 256 features, how many eigen vectors does it provide? What will be the size of eigen vectors?
2. How eigen vectors of AA^T are obtained from the eigen vectors of A^TA?
3. What is the specialty of Eigen vectors?

Answers

1. 30 vectors each of size 30
2. By multiplying it with the observation matrix
3. These vectors when multiplied with the matrix does not change its direction

Feature Vectors



Obtaining Feature Vectors

- u_k represent eigen face
- They form basis images
- Feature vectors in reduced dimension is obtained through linear projection of mean subtracted face image onto eigen vectors

$$w_{ik} = u_k^t \, \phi_i \Big| \, k = 1,2 \ldots M - 1 \tag{1}$$

Image Γ_i is represented by M-1 columnar vector denoted by w_i

$$w_i = \left[w_{i_1}, w_{i_{2},\ldots} w_{iM-1}\right]^t$$

Face Recognition Using PCA – Inference Phase

- Verification Mode
- Person provides an image $\tilde{I}$ and claims the identity
- Find feature vector for acquired image

$$\widetilde{w_k} = u_k^t\left(\tilde{\Gamma} - \psi\right)k = 1,2, \ldots M - 1$$

Compare the distance between acquired vector and claimed identity . Distance is within a particular threshold, then the person is granted access

Pause Moment

- Self Evaluate
  - Given vector $\tilde{w} = (1,2,3,4)$ and reference vector w for the claimed identity is (1,4,2,1), Will a Manhattan distance based client model authorize the individual with a threshold of 4?

Answers = mod(1-1) + mod(2-4) + mod(3-2) + mod(4-1) = 0+ 2+ 1 + 3 = 6. Since it exceeds threshold, user will not be authorized

Questions

- Typically, 5 to 6 sample images are captured for same person. How samples are used in the recognition process?
- What will you do to visualize the reduced image given a feature vector?
- Does PCA really decorrelate data? How to verify that?

Simulation of PCA

```python
import numpy as np
x = [1,2,3,4,5,6,7,8,9,10]
y = [1,2,3,4,5,6,7,8,9,10]
z = [14,11,21,56,1,76,34,2,11,10]

A = np.zeros(shape=(10,3))
A[:,0] = x
A[:,1] = y
A[:,2] = z
```

x, y, z are features stacked together into observation matrix A

```python
#Compute Mean
psi = np.mean(A, axis=0)
#Mean subtract data
A = A - psi
#Find Covariance of matrix
C = np.cov(np.transpose(A))
```

```
[[  9.16666667   9.16666667  -6.77777778]
 [  9.16666667   9.16666667  -6.77777778]
 [ -6.77777778  -6.77777778 609.15555556]]
```

```
[[ 9.16666667 9.16666667 -6.77777778]
[ 9.16666667 9.16666667 -6.77777778]
[ -6.77777778 -6.77777778 609.15555556]]
```

Contd..

```
from numpy import linalg as LA
w, v = LA.eig(C)
```

W

(Eigen Values)

[-5.32907052e-15
1.81778680e+01
6.09311021e+02]

V **(Eigen Vectors)**

[[-7.07106781e-01 7.07013792e-01 -1.14672459e-02]
[7.07106781e-01 7.07013792e-01 -1.14672459e-02]
[-6.41847686e-17 1.62171347e-02 9.99868494e-01]]

Contd..

```python
v = v[:,1:3]
W = np.dot(A,v)
```

Data points in reduced feature space

```
[[ -6.51880862 -9.49553233]
 [ -5.15343244 -12.5180723 ]
 [ -3.57723351 -2.54232185]
 [ -1.59560621 32.43014093]
 [ -1.07352104 -22.58556071]
 [ 1.55679165 52.38164182]
 [ 2.28969958 10.3642306 ]
 [ 3.18477885 -21.65449569]
 [ 4.74476065 -12.67861374]
 [ 6.1425711 -13.70141673]]
```

```python
C = np.cov(np.transpose(W))
```

Covariance Matrix in reduced dimension space

```
[[ 1.81778680e+01 -5.57676958e-14]
 [-5.57676958e-14 6.09311021e+02]]
```

Pause Moment

- Self Evaluate
  - How do you interpret the covariance matrix?

Answers

Covariance matrix is constructed between every pair of features, +ve value indicate positive correlation, -ve value indicate negative correlation , any value close to zero indicate uncorrelated features.

In PCA , the resulting dimensions are uncorrelated.

References

- **http://vision.ucsd.edu/~iskwak/ExtYaleDatabase/Yale%20Face%20Database.htm**
- **All the images are non – licensed images free for commercial distribution.**
- **https://sites.cs.ucsb.edu/~mturk/Papers/mturk-CVPR91.pdf (Turk and Pentland; "Face Recognition using Eigen Faces")**
- **Amine Nait Ali, "Signal and Image Processing for Biometrics "**