- New concepts:
  - Images as vectors
  - Interpreting column space, null space for a collection of faces
  - Face recognition by projection

- `facedata.zip` is the MIT face database. Unzip: obtain directory 'rawdata'
- These Matlab instructions display one face within the MIT face collection numbered from 1223 to 5222 (3993 faces, some indices skipped), each face is represented by a vector with $128^2 = 16384$ gray-scale values.
- Faces are displayed using the function

```
function shwface(a)
 face=reshape(a,128,128)';
 imagesc(face); colormap(gray(256));
 axis('equal')
end
```

```
matlab>>dir=strcat(getenv("HOME"),"/courses/MATH347");
        fid=fopen(strcat(dir,"/rawdata/1223"));
        face1=fread(fid);
        fclose(fid);
        figure(1); shwface(face1);
```

```
matlab>>
```

- `facedata.zip` is the MIT face database. Unzip: obtain directory 'rawdata'
- These Matlab instructions display one face within the MIT face collection numbered from 1223 to 5222 (3993 faces, some indices skipped), each face is represented by a vector with $128^2 = 16384$ gray-scale values.
- Faces are displayed using the function

```
function shwface(a)
 face=reshape(a,128,128)';
 imagesc(face); colormap(gray(256));
 axis('equal')
end
```

```
matlab>>dir=strcat(getenv("HOME"),"/courses/MATH347");
        fid=fopen(strcat(dir,"/rawdata/1223"));
        face1=fread(fid);
        fclose(fid);
        figure(1); shwface(face1);
```

```
>>
```

```
matlab>>
```

- Read in another face

```
matlab>>dir=strcat(getenv("HOME"),"/courses/MATH347");
        fid=fopen(strcat(dir,"/rawdata/1323"));
        face2=fread(fid);
        fclose(fid);
        figure(2); shwface(face2);
```

- Face vectors are not orthogonal, there is overlap in the face information

```
matlab>>180*acos(face1'*face2/norm(face1)/norm(face2))/pi
```

- $\theta = 0 \Rightarrow$ colinear (identical face features)
- $\theta = 90^\circ \Rightarrow$ orthogonal (completely different, not likely for members of the same species!)

- Read in another face

```
matlab>>dir=strcat(getenv("HOME"),"/courses/MATH347");
        fid=fopen(strcat(dir,"/rawdata/1323"));
        face2=fread(fid);
        fclose(fid);
        figure(2); shwface(face2);
```

>>

- Face vectors are not orthogonal, there is overlap in the face information

```
matlab>>180*acos(face1'*face2/norm(face1)/norm(face2))/pi
```

- $\theta = 0 \Rightarrow$ colinear (identical face features)
- $\theta = 90^\circ \Rightarrow$ orthogonal (completely different, not likely for members of the same species!)

- Read in another face

```
matlab>>dir=strcat(getenv("HOME"),"/courses/MATH347");
        fid=fopen(strcat(dir,"/rawdata/1323"));
        face2=fread(fid);
        fclose(fid);
        figure(2); shwface(face2);
```

>>

- Face vectors are not orthogonal, there is overlap in the face information

```
matlab>>180*acos(face1'*face2/norm(face1)/norm(face2))/pi
```

>>35.109

- $\theta = 0 \Rightarrow$ colinear (identical face features)
- $\theta = 90^\circ \Rightarrow$ orthogonal (completely different, not likely for members of the same species!)

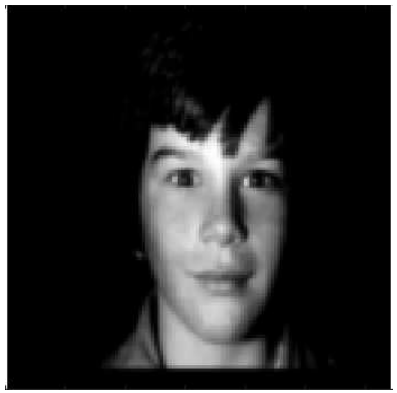
- The collection is gathered into a matrix $\boldsymbol{F} = [\ \boldsymbol{f}_1 \ \ \ldots \ \ \boldsymbol{f}_{3993}\ ] \in \mathbb{R}^{16384 \times 3993}$
- An "average" face vector $\boldsymbol{a}$ is obtained by

$$\boldsymbol{a} = \frac{1}{3993} \sum_{j=1}^{3993} \boldsymbol{f}_j$$

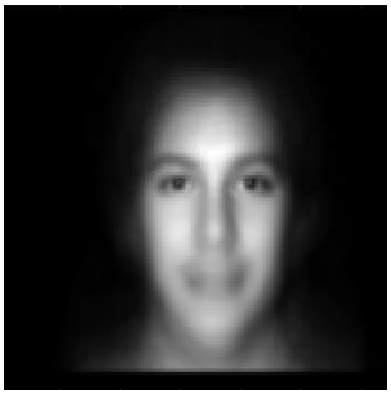
```
matlab>>data=strcat(dir,"/faces.mat"); load(data);
        figure(3); shwface(a)
```

- Construct a matrix of deviations from the average $\boldsymbol{D} = [\ \boldsymbol{d}_1 \ \ \ldots \ \ \boldsymbol{d}_{3993}\ ]$, $\boldsymbol{d}_j = \boldsymbol{f}_j - \boldsymbol{a}$

Face 1223



Face 1323



Average face

- The collection is gathered into a matrix $\boldsymbol{F} = [\ \boldsymbol{f}_1 \ \ \ldots \ \ \boldsymbol{f}_{3993}\ ] \in \mathbb{R}^{16384 \times 3993}$
- An "average" face vector $\boldsymbol{a}$ is obtained by

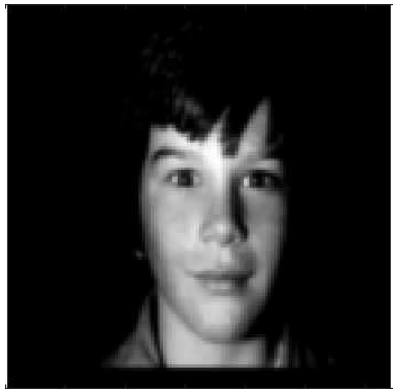
$$\boldsymbol{a} = \frac{1}{3993} \sum_{j=1}^{3993} \boldsymbol{f}_j$$

```
matlab>>data=strcat(dir,"/faces.mat"); load(data);
        figure(3); shwface(a)
```

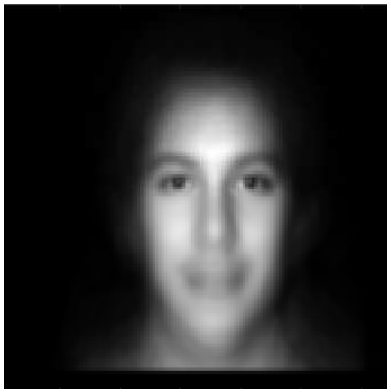
```
>>
```

- Construct a matrix of deviations from the average $\boldsymbol{D} = [\ \boldsymbol{d}_1 \ \ \ldots \ \ \boldsymbol{d}_{3993}\ ]$, $\boldsymbol{d}_j = \boldsymbol{f}_j - \boldsymbol{a}$

Face 1223



Face 1323



Average face

- Seek an economical way to reconstruct a face

$$\boldsymbol{f} = \boldsymbol{A}\boldsymbol{r} + \boldsymbol{a}$$

  - $\boldsymbol{f}$ reconstructed face
  - $\boldsymbol{a}$ average face
  - $\boldsymbol{A} \in \mathbb{R}^{128^2 \times n}$ a subcollection of n faces
- How to choose $\boldsymbol{A}$?
- Of all deviations from the average face choose the closest to orthogonal to $\boldsymbol{a}$

$$\theta_j = \operatorname{acos}\left(\frac{\boldsymbol{d}_j^T \boldsymbol{a}}{\|\boldsymbol{d}_j\| \, \|\boldsymbol{a}\|} \right)$$

  Choose $\boldsymbol{b}_1 = \boldsymbol{d}_k$ with k the index for which $|\theta_k - \pi/2| \leqslant |\theta_j - \pi/2|$ for all $j \in \{1, 2, \ldots, 3993\}$
- Then choose $\boldsymbol{b}_2$ closest to orthogonal to both $\boldsymbol{a}$ and $\boldsymbol{b}_1$
- The matrix $\boldsymbol{B} = [\ \boldsymbol{b}_1 \ \ \ldots \ \ \boldsymbol{b}_n\ ]$ does not necessarily have orthonormal columns
- Apply Gram-Schmidt to $\boldsymbol{B}$ and obtain $\boldsymbol{A}$ with orthonormal columns

- A face can be approximately reconstructed by linear combination

$$\boldsymbol{f} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{a}$$

Original face image

Reconstructed face with $n = 99$

- How close is the reconstruction? Find the angle

```
matlab>>f=A*R(:,1)+a; acos(face1'*f/norm(face1)/norm(f))*180/pi
```

- Obtain an angle of $\theta = 10.7°$ fairly close to perfect reconstruction ($\theta = 0$)
- Better reconstruction accuracy would require increased n

- A face can be approximately reconstructed by linear combination

$$\boldsymbol{f} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{a}$$

Original face image

Reconstructed face with $n = 99$

- How close is the reconstruction? Find the angle

```
matlab>>f=A*R(:,1)+a; acos(face1'*f/norm(face1)/norm(f))*180/pi
```

```
>>10.7081
```

- Obtain an angle of $\theta = 10.7°$ fairly close to perfect reconstruction ($\theta = 0$)
- Better reconstruction accuracy would require increased n