Topic 07:

Linear Subspace Models

• Subspace appearance models
• Representing (discrete) images as vectors
• Choosing an appropriate basis
• Principal Component Analysis (PCA)
• PCA for subspace modeling
**Question:** Suppose we have a dataset of scaled, aligned images of human eyes. How can we find an efficient representation of them?

**Generative Model:** For example, suppose we can approximate each image in the data set with a parameterized model of the form

\[ I(\tilde{x}) \approx g(\tilde{x}, \tilde{a}) , \]

where \( \tilde{a} \) is a (low-dimensional) vector of coefficients.

**Possible uses:**

- reduce the dimension of the data set (compression)
- generate novel instances (density estimation)
- (possibly) detection/recognition
subspace appearance models

**Idea:** Images are not random, especially those of an object, or similar objects, under different viewing conditions.

Rather, than storing every image, we might try to represent the images more effectively, e.g., in a lower dimensional *subspace*. 
linear subspace models

**Goal:** Explore linear models of data.

**Motivation:** A central question in vision concerns how we represent a collection of data vectors, such as images of an object under a wide range of viewing conditions (lighting and viewpoints).

- We consider the construction of low-dimensional bases for an ensemble of training data using principal components analysis (PCA).

- We introduce PCA, its derivation, its properties, and some of its uses.

- We briefly critique its suitability for object detection.
linear subspace appearance models

**Idea:** Images are not random, especially those of an object, or similar objects, under different viewing conditions.

Rather, than storing every image, we might try to represent the images more effectively, e.g., in a lower dimensional *subspace*.

How do we find a low-dimensional basis to accurately model (approximate) each image of the training ensemble (as a linear combination of basis images)?
linear subspace appearance models

Idea: Images are not random, especially those of an object, or similar objects, under different viewing conditions.

Rather, than storing every image, we might try to represent the images more effectively, e.g., in a lower dimensional subspace.
Topic 07:

Linear Subspace Models

• Subspace appearance models
• Representing (discrete) images as vectors
• Choosing an appropriate basis
• Principal Component Analysis (PCA)
• PCA for subspace modeling
representing images as vectors

Image $X$ (2 pixels):

\[
\begin{bmatrix}
50 \\
255
\end{bmatrix}
\]

Image $X$ is just a 2-dimensional vector.

Image $X$ (3 pixels):

\[
\begin{bmatrix}
50 \\
255 \\
30
\end{bmatrix}
\]

Image $X$ is just a 3-dimensional vector.
effect of changing the basis

Case A: Pixel intensities are uncorrelated

\[ X_i = \begin{bmatrix} x_i^1 \\ x_i^2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} x_i^1 + \begin{bmatrix} 1 \\ 0 \end{bmatrix} x_i^2 \]

Case B: Pixel intensities are correlated

\[ X_i = y_i^1 B_1 + y_i^2 B_2 \approx y_i^1 B_\perp \]

possibly small
changing the basis: matrix notation

For one training image:

\[ X_i = \begin{bmatrix} 1 & 0 \end{bmatrix} X_i^1 + \begin{bmatrix} 0 & 1 \end{bmatrix} X_i^2 \]

basis matrix

\[ X_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_i^1 \\ X_i^2 \end{bmatrix} \]

coordinate vector

\[ X_i = \begin{bmatrix} B_1 & B_2 \end{bmatrix} \begin{bmatrix} y_i^1 \\ y_i^2 \end{bmatrix} \]

\( j \)-th column is the \( j \)-th basis vector

All \( N \) training images:

\[
\begin{bmatrix}
X_1 & X_2 & \cdots & X_N
\end{bmatrix} =
\begin{bmatrix}
B_1 & B_2 \\
\vdots & \vdots \\
B_1 & B_2
\end{bmatrix}
\begin{bmatrix}
y_1^1 & y_1^2 & \cdots & y_1^N \\
y_2^1 & y_2^2 & \cdots & y_2^N \\
\vdots & \vdots & \ddots & \vdots \\
y_N^1 & y_N^2 & \cdots & y_N^N
\end{bmatrix}
\]
changing the basis: matrix notation

\[ X_i = \begin{bmatrix} 1 \\ 0 \end{bmatrix} X_i^1 + \begin{bmatrix} 0 \\ 1 \end{bmatrix} X_i^2 \]

All \( N \) images

\[
\begin{bmatrix} X_1 & X_2 & \cdots & X_N \end{bmatrix} = \begin{bmatrix} B_1 & B_2 & \cdots & B_m \end{bmatrix} \begin{bmatrix} y_1^1 & y_2^1 & \cdots & y_N^1 \\
 y_1^2 & y_2^2 & \cdots & y_N^2 \\
 \vdots & \vdots & \ddots & \vdots \\
 y_1^M & y_2^M & \cdots & y_N^M \end{bmatrix}
\]

For \( M \)-dimensional images:

\[
\begin{bmatrix} X_1 & X_2 & \cdots & X_N \end{bmatrix} = \begin{bmatrix} B_1 & B_2 & \cdots & B_m \end{bmatrix} \begin{bmatrix} y_1 & y_2 & \cdots & y_N \\
 y_1 & y_2 & \cdots & y_N \\
 \vdots & \vdots & \ddots & \vdots \\
 y_1 & y_2 & \cdots & y_N \end{bmatrix}
\]
Topic 07:

Linear Subspace Models

• Subspace appearance models
• Representing (discrete) images as vectors
• Choosing an appropriate basis
• Principal Component Analysis (PCA)
• PCA for subspace modeling
choosing an appropriate basis

Case A: Pixel intensities are uncorrelated

Case B: Pixel intensities are correlated

Idea #1: When pixel intensities are related, it may be possible to express an image in terms of basis vectors where only a few of the coordinates are significant (i.e., not close to 0)

\[ x_i = B_1 \cdot y_1 + B_2 \cdot y_2 \]
choosing an optimal basis

Idea #2: In real photos, the pixel intensities are highly correlated

Algorithm:

1. Find optimal set of basis vectors $B_1, \ldots, B_m$ (a.k.a. Principal Components)
2. Compute image coordinates in that basis
3. Discard the axes with near-zero coordinates for all images

Idea #1: When pixel intensities are related, it may be possible to express an image in terms of basis vectors where only a few of the coordinates are significant (i.e. not close to 0)

$$X_i = B_1 \cdot y_i^1 + B_2 \cdot y_i^2$$
Goal of Principal Component Analysis (PCA)

\[ X_i = [1] X_i^1 + [0] X_i^2 \]

Goal: Given \( d \), choose the most significant basis vectors, \( B_1, \ldots, B_d \)

\[
\begin{bmatrix}
X_1 & X_2 & \cdots & X_N
\end{bmatrix} =
\begin{bmatrix}
B_1 & B_2 & \cdots & B_m
\end{bmatrix} \cdot
\begin{bmatrix}
y_1^1 & y_2^1 & \cdots & y_N^1 \\
y_1^2 & y_2^2 & \cdots & y_N^2 \\
y_1^d & y_2^d & \cdots & y_N^d \\
\vdots & \vdots & \ddots & \vdots \\
y_1^M & y_2^M & \cdots & y_N^M \\
\end{bmatrix}
\]
goal of Principal Component Analysis (PCA)

\[ X_i = [1] X_i^1 + [0] X_i^2 \]

Reconstruction error for dimensions

\[ \begin{bmatrix} X_1 & X_2 & \ldots & X_N \end{bmatrix} - \begin{bmatrix} B_1 & B_2 & \ldots & B_d \end{bmatrix} \cdot \begin{bmatrix} y_1^1 & y_2^1 & \cdots & y_N^1 \\ y_1^d & y_2^d & \cdots & y_N^d \end{bmatrix} \]

chosen optimally

\[ \text{Goal: Find } B_1, \ldots, B_m \text{ that minimize the squared reconstruction error} \]

\[ \sum_{i=1}^{N} \min_{y_i^1, \ldots, y_i^d} \| X_i - [B_1 \ldots B_d] [y_i^1 \ldots y_i^d] \|^2 \]
The goal of Principal Component Analysis (PCA) is to find a new set of coordinates that represent the data in a lower-dimensional space while preserving as much of the variance as possible. This is achieved by transforming the original data into a new coordinate system where the first axis (or principal component) is the direction of the highest variance, and subsequent axes are orthogonal to and of decreasing variance.

Mathematically, this can be represented as:

\[
X_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} X'_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} X''_i
\]

Where \( X_i \) is the original data point, \( X'_i \) is the projection onto the first principal component, and \( X''_i \) is the projection onto the second principal component.

Optimal d-dimensional representation of the training images:

\[
\begin{bmatrix}
B_1 & B_2 & \cdots & B_d \\
y_1^1 & y_2^1 & \cdots & y_N^1 \\
y_1^d & y_2^d & \cdots & y_N^d
\end{bmatrix}
\]

**Goal:** Find \( B_1, \ldots, B_m \) that minimize the squared reconstruction error:

\[
\sum_{i=1}^{N} \min_{y^i} \| X_i - [B_1 \ldots B_d][y_i^1 \ldots y_i^d] \|^2
\]
Topic 07:

Linear Subspace Models

- Subspace appearance models
- Representing (discrete) images as vectors
- Choosing an appropriate basis
- Principal Component Analysis (PCA)
- PCA for subspace modeling
Input: d, matrix \( X \)
Output: Vectors \( B_1, \ldots, B_d \)

1. Compute the average
   \[
   \overline{X} = \frac{1}{N} \sum X_i
   \]

2. Subtract the average from each \( X_i \):
   \[
   Z_i = X_i - \overline{X}
   \]

3. Define \( Z = [Z_1, \ldots, Z_N] \)

4. \( B_1, \ldots, B_d \) = eigenvectors of matrix \( ZZ^T \) with the \( d \) largest eigenvalues
Principal Component Analysis: Step 1

**Input:** $d$, matrix $X$

**Output:** Vectors $B_1, \ldots, B_d$

1. Compute the average patch: $\bar{X} = \frac{1}{N} \sum X_i$

2. Subtract the average from each $X_i$: $Z_i = X_i - \bar{X}$

3. Define $Z = [Z_1, \ldots, Z_N]$.

4. $B_1, \ldots, B_d =$ eigenvectors of matrix $ZZ^T$ with the $d$ largest eigenvalues.
Principal Component Analysis: Step 2

Input: $d$, matrix $X$
Output: Vectors $B_1, \ldots, B_d$

1. Compute the average patch: $\bar{X} = \frac{1}{N} \sum X_i$

2. Subtract the average from each $X_i$: $Z_i = X_i - \bar{X}$

3. Define $Z = [Z_1, \ldots, Z_N]$

4. $B_1, \ldots, B_d$ = eigenvectors of matrix $ZZ^T$ with the $d$ largest eigenvalues
Principal Component Analysis: Step 4

Input: $d$, matrix $X$
Output: Vectors $B_1, \ldots, B_d$

1. Compute the average patch: $\bar{X} = \frac{1}{N} \sum X_i$
2. Subtract the average from each $X_i$: $Z_i = X_i - \bar{X}$
3. Define $Z = [Z_1, \ldots, Z_N]$
4. $B_1, \ldots, B_d$ = eigenvectors of matrix $ZZ^T$ with the $d$ largest eigenvalues
1\textsuperscript{st} principal component of axis-aligned data

For the Standard Basis we have

\[ B_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad B_m = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \]

\[ \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_N \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \vdots \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} r_1^1 \\ r_2^1 \\ \vdots \\ r_N^1 \end{bmatrix} \]

\[ \text{Variance of } j\text{-th row of } Z: \]

\[ G_j^2 = \frac{1}{N} \sum_{i=1}^{N} (Z_i^j)^2 \]

\[ G_j^2 = \frac{1}{N} \mathbf{r}_j \cdot (\mathbf{r}_j)^T \]

\[ \text{In matrix notation} \]

\[ \begin{bmatrix} Z_1^1 & Z_2^1 & \cdots & Z_N^1 \\ Z_1^2 & Z_2^2 & \cdots & Z_N^2 \\ \vdots & \vdots & \ddots & \vdots \\ Z_1^M & Z_2^M & \cdots & Z_N^M \end{bmatrix} = \begin{bmatrix} \mathbf{r}_1^1 \\ \mathbf{r}_2^1 \\ \vdots \\ \mathbf{r}_N^1 \end{bmatrix} \]

\[ \text{not very significant} \quad (G_2^2 \ll G_1^2) \]

\[ \text{significant} \quad (G_1^2 \text{ large}) \]

\[ \text{"centered" intensity of pixel } z \]

\[ \text{"centered" intensity of pixel } z \]
1\textsuperscript{st} principal component of axis-aligned data

So when sample distribution is axis aligned

\[ \Rightarrow \]

the first basis vector should be the axis that maximizes variance

\[ \text{Var}(r_j) = r_j (r_j)^T \]

\begin{itemize}
  \item Variance of j-th row of Z:
    \[ G_j^2 = \frac{1}{N} \sum_{i=1}^{N} (Z_{i,j})^2 \]
  \item In matrix notation
    \[ G_j^2 = \frac{1}{N} r_j (r_j)^T \]
\end{itemize}

significant \( (G_1^2 \text{ large}) \)

not very significant \( (G_2^2 \ll G_1^2) \)

"centered" intensity of pixel 2

"centered" intensity of pixel 1
1\textsuperscript{st} principal component in general case

So when sample distribution is \textbf{NOT} axis aligned

\[ \Rightarrow \]

must find unit vector $B_1$

yielding maximum variance

1. Image coordinates along $B_1$ given by

\[
\begin{bmatrix}
Z_1^T \\
Z_2^T \\
\vdots \\
Z_N^T
\end{bmatrix}
\begin{bmatrix}
B_1
\end{bmatrix}
\]

project each image along direction $B_1$

In matrix notation

\[
6_j^2 = \frac{1}{N} r_j . (r_j)^T
\]
1st principal component in general case

So when sample distribution is NOT axis aligned

⇒ must find unit vector $B_1$

yielding maximum variance

1. Image coordinates along $B_1$ given by

$$\begin{bmatrix}
Z_1^T \\
Z_2^T \\
\vdots \\
Z_N^T
\end{bmatrix} B_1$$

project each image along direction $B_1$

2. Variance of those coordinates given by

$$B_1^T \begin{bmatrix} Z_1 & Z_2 & \cdots & Z_N \end{bmatrix} \begin{bmatrix} Z_1^T \\
Z_2^T \\
\vdots \\
Z_N^T
\end{bmatrix} B_1 \frac{1}{N}$$

"centered" intensity of pixel 2

$Z_i$

"centered" intensity of pixel 1
1st principal component in general case

1. Project each image along direction $B_i$

$$\begin{bmatrix} Z_1^T \\ Z_2^T \\ \vdots \\ Z_n^T \end{bmatrix} B_i$$

2. Variance of those coordinates given by

$$B_i^T (Z Z^T) B_i$$

Using Lagrange multipliers we maximize

$$B_i^T (Z Z^T) B_i + \lambda (B_i^T B_i - 1)$$

3. Must find unit $B_i$ that maximizes variance

$$B_i = \text{argmax} \{B^T (Z Z^T) B | B^T B = 1\}$$

4. Yielding maximum variance

$$B_i = \text{argmax} \{B^T (Z Z^T) B | B^T B = 1\}$$

So when sample distribution is not axis aligned
1st principal component in general case

5. Differentiating wrt $B_1$ and setting to zero:

$$(ZZ^T)B_1 - \lambda B_1 = 0 \iff (ZZ^T)B_1 = \lambda B_1$$

1. Image coordinates along $B_1$ given by:

$$\begin{bmatrix} Z_1^T \\ Z_2^T \\ \vdots \\ Z_N^T \end{bmatrix} B_1$$

Project each image along direction $B_1$. We maximize:

$$B_1^T (ZZ^T) B_1 + \phi(B_1^T B_1 - 1)$$

2. Variance of those coordinates given by:

$$B_1^T \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_N \end{bmatrix} \begin{bmatrix} Z_1^T \\ Z_2^T \\ \vdots \\ Z_N^T \end{bmatrix} B_1 \frac{1}{N}$$

3. Must find unit $B_1$ that maximizes

$$B_1^T (ZZ^T) B_1$$
Result: $B_1$ is the eigenvector of $ZZ^T$ with largest eigenvalue

$$(ZZ^T)B_1 = \lambda B_1$$

1. Image coordinates along $B_1$ given by

$$\begin{bmatrix} Z_1^T \\ Z_2^T \\ \vdots \\ Z_N^T \end{bmatrix} B_1$$

project each image along direction $B_1$

2. Variance of those coordinates given by

$$B_1^T \begin{bmatrix} Z_1 & Z_2 & \cdots & Z_N \end{bmatrix} \begin{bmatrix} Z_1^T \\ \vdots \\ Z_N^T \end{bmatrix} B_1 \frac{1}{N}$$

3. Must find unit $B_1$ that maximizes

$$B_1^T (ZZ^T) B_1$$

4. Using Lagrange multipliers we maximize

$$B_1^T (ZZ^T) B_1 + \alpha (B_1^T B_1 - 1)$$
computing all principal components

**Theorem:** (Minimum reconstruction error) The orthogonal basis \( B \), of rank \( K < N^2 \), that minimizes the squared reconstruction error over training data, \( \{ \vec{I}_l \}_{l=1}^L \), i.e.,

\[
\sum_{l=1}^{L} \min_{\vec{a}_l} \| \vec{I}_l - B \vec{a}_l \|_2^2
\]

is given by the first \( K \) eigenvectors of the data covariance matrix

\[
C = \frac{1}{L} \sum_{l=1}^{L} \vec{I}_l \vec{I}_l^T \in \mathbb{R}^{N^2 \times N^2}, \quad \text{for which} \quad C U = U D
\]

where \( U = [\vec{u}_1, ..., \vec{u}_{N^2}] \) is orthogonal, and \( D = \text{diag}(d_1, ..., d_{N^2}) \) with \( d_1 \geq d_2 \geq ... \geq d_{N^2} \).

That is, the optimal basis vectors are \( \vec{b}_k = \vec{u}_k \), for \( k = 1 ... K \). The corresponding basis images \( \{ b_k(\vec{x}) \}_{k=1}^K \) are often called eigen-images.
properties of PCA: maximum variance

**Maximum Variance:** PCA also gives the $K$-D subspace that captures the greatest fraction of the total variance in the training data.

- For $a_1 = \tilde{b}_1^T \tilde{I}$, the direction $\tilde{b}_1$ that maximizes the coef variance $E[a_1^2] = \tilde{b}_1^T \tilde{C} \tilde{b}_1$, s.t. $\tilde{b}_1^T \tilde{b}_1 = 1$, is the first eigenvector of $\tilde{C}$.
- The second maximizes $\tilde{b}_2^T \tilde{C} \tilde{b}_2$ subject to $\tilde{b}_2^T \tilde{b}_2 = 1$ and $\tilde{b}_1^T \tilde{b}_2 = 0$.
- For $a_k = \tilde{b}_k^T \tilde{I}$, and $\tilde{a} = (a_1, \ldots, a_K)$, the subspace coefficient covariance is $E[\tilde{a} \tilde{a}^T] = \text{diag}(d_1, \ldots, d_K)$. That is, the diagonal entries of $\tilde{D}$ are marginal variances of the subspace coefficients:
  $$\sigma_k^2 \equiv E[a_k^2] = d_k$$

So the total variance captured in the subspace is sum of first $K$ eigenvalues of $\tilde{C}$.

- Total variance lost owing to the subspace projection (i.e., the out-of-subspace variance) is the sum of the last $N^2 - K$ eigenvalues:
  $$\frac{1}{L} \sum_{l=1}^L \left[ \min_{\tilde{a}_l} \| \tilde{I}_l - \tilde{B} \tilde{a}_l \|_2^2 \right] = \sum_{k=K+1}^{N^2} \sigma_k^2$$
**Decorrelated Coefficients:** C is diagonalized by its eigenvectors, so D is diagonal, and the subspace coefficients are uncorrelated.

- Under a Gaussian model of the images (where the images are drawn from an $N^2$-dimensional Gaussian pdf), this means that the coefficients are also statistically independent.
Topic 07: Linear Subspace Models

- Subspace appearance models
- Representing (discrete) images as vectors
- Choosing an appropriate basis
- Principal Component Analysis (PCA)
- PCA for subspace modeling
eye subspace model

Subset of 1196 eye images (25 × 20):

Left Eyes

Right Eyes

With $V_k \equiv \sum_{j=1}^{k} \sigma_j^2$,

- $dQ_k \equiv \sigma_k^2/V_L$ is the fraction of total variance contributed by the $k^{th}$ principal component (left)

- $Q_k \equiv V_k/V_L$ is the fraction of total variance captured by the subspace formed from the first $k$ principal components (right)
eye subspace model

Mean Eye:

Basis Images (1−6, and 10:5:35):

Reconstructions (for $K = 5, 20, 50$):
Generative eye model

Generative model, $\mathcal{M}$, for random eye images:

$$\bar{I} = \bar{m} + \left( \sum_{k=1}^{K} a_k \bar{b}_k \right) + \bar{e}$$

where $\bar{m}$ is the mean eye image, $a_k \sim \mathcal{N}(0, \sigma_k^2)$, $\sigma_k^2$ is the sample variance associated with the $k^{th}$ principal direction in the training data, and $\bar{e} \sim \mathcal{N}(0, \sigma_e^2 I_{N^2})$ where $\sigma_e^2 = \frac{1}{N^2} \sum_{k=K+1}^{N^2} \sigma_k^2$ is the per pixel out-of-subspace variance.

Random Eye Images:
Generative model, $\mathcal{M}$, for random eye images:

$$\bar{I} = \bar{m} + \left( \sum_{k=1}^{K} a_k \bar{b}_k \right) + \bar{e}$$

where $\bar{m}$ is the mean eye image, $a_k \sim \mathcal{N}(0, \sigma_k^2)$, $\sigma_k^2$ is the sample variance associated with the $k^{th}$ principal direction in the training data, and $\bar{e} \sim \mathcal{N}(0, \sigma_e^2 I_{N^2})$ where $\sigma_e^2 = \frac{1}{N^2} \sum_{k=K+1}^{N^2} \sigma_k^2$ is the per pixel out-of-subspace variance.

So the likelihood of an image of an eye given this model $\mathcal{M}$ is

$$p(\bar{I} | \mathcal{M}) = \left( \prod_{k=1}^{K} p(a_k | \mathcal{M}) \right) p(\bar{e} | \mathcal{M})$$

where

$$p(a_k | \mathcal{M}) = \frac{1}{\sqrt{2\pi} \sigma_k} e^{-\frac{a_k^2}{2\sigma_k^2}} , \quad p(\bar{e} | \mathcal{M}) = \prod_{j=1}^{N^2} \frac{1}{\sqrt{2\pi} \sigma_e} e^{-\frac{e_j^2}{2\sigma_e^2}} .$$
eigenfaces (Turk & Pentland, 1991)

Shown below is the model learned from a collection of frontal faces, normalized for contrast, scale, and orientation, with the backgrounds removed prior to PCA.

Here are the mean image (upper-left) and the first 15 eigen-images. The first three show strong variations caused by illumination. The next few appear to correspond to the occurrence of certain features (hair, hairline, beard, clothing, etc).
eigen-faces: example dataset
the top 6 principal components
eigenface representation (with d=3)

\[ X_1 \ (M \ dimensions) \]

\[ X_1 \ (d-dimensional \ approximation \ d=3) \]

\[ X \]

\[ y_1^* \]

\[ B_1 \]

\[ y_2^* \]

\[ B_2 \]

\[ y_3^* \]

\[ B_3 \]
The generative model:

- PCA finds the subspace (of a specified dimension) that maximizes (projected) signal variance.

- A single Gaussian model is naturally associated with a PCA representation. The principal axes are the principal directions of the Gaussian’s covariance.

- This can be a simple way to find low-dimensional representations that are effective as prior models and for optimization.

Issues:

- The single Gaussian model is often rather crude. PCA coefficients often exhibit significantly more structure (cf. Murase & Nayar).

- As a result of this unmodelled structure, detectors based on single Gaussian models are often poor. (see the Matlab tutorial detectEigenEyes.m)
appearance manifolds in PCA subspace

Murase and Nayar (1995)
- images of multiple objects, taken from different positions on the viewsphere and with different lighting directions
- each object occupies a manifold in the subspace (as a function of position on the viewsphere and lighting direction)
- recognition: nearest neighbour assuming dense sampling of object pose variations in the training set.