Component Analysis for Computer Vision

Fernando De la Torre
Carnegie Mellon
THE ROBOTICS INSTITUTE

Tutorial European Conference on Computer Vision
May 2006

Outline

• Introduction
• Generative models
  – Principal Component Analysis (PCA).
  – Non-negative Matrix Factorization (NMF).
  – Independent Component Analysis (ICA).
• Discriminative models
  – Linear Discriminant Analysis (LDA).
  – Oriented Component Analysis (OCA).
  – Canonical Correlation Analysis (CCA).
  – Relevant Component Analysis (RCA).
• Standard extensions of linear models
  – Latent variable models.
  – Tensor factorization.
  – Kernel methods.

Component Analysis

• Computer Vision & Image Processing
  – Structure from motion.
  – Spectral graph methods for segmentation.
  – Appearance and shape models.
  – Fundamental matrix estimation and calibration.
  – Compression.
  – Classification.
  – Dimensionality reduction and visualization.
• Signal Processing
  – Spectral estimation, system identification (e.g. Kalman filter), sensor
    array processing (e.g. cocktail problem, eco cancellation), blind source
    separation, ...
• Computer Graphics
  – Compression (BRDF), synthesis, ...
• Speech, bioinformatics, combinatorial problems.

Why Subspace Methods?

• Learning: High dimensional data lie in a low
dimensional manifold.
• Estimation: Many cv problems (SFM, calibration, ...) can be posed as subspace
  estimation.
  • Better generalization (noise removal).
  • Simple parameterization.
  • Lower computational complexity.
  • Closed form solution and global minimum.
Generative Models

\[ D \approx BC \]

- Principal Component Analysis/Singular Value Decomposition.
  1) Robust PCA/SVD.
  2) PCA with uncertainty and missing data.
  3) Parameterized PCA.
  4) PCA over continuous spaces.
  5) Filtered PCA.
  6) PCA of rotated images.
  7) Mixture of subspaces.
- Non-Negative Matrix Factorization.
  8) PCA and NMF for spectral clustering.
- Independent Component Analysis.

Principal Component Analysis (PCA)

(Pearson, 1901; Hotelling, 1933; Mardia et al., 1979; Jolliffe, 1986; Diamantaras, 1996)

• PCA finds the directions of maximum variation of the data based on linear correlation.
• PCA decorrelates the original variables.

Snap-shot method & SVD

- If \( d \gg n \) (e.g., images 100*100 vs. 300 samples) no \( DD^T \).
- \( DD^T \) and \( D^T D \) have the same eigenvalues (energy) and related eigenvectors (by \( D \)).
- \( B \) is a linear combination of the data! \( (\text{Sirovich, 1987})\)
\[ DD^T B = BA \quad B = Da \quad D^T D \alpha = D^T _a \alpha \Lambda \]
\[ [\alpha, \lambda] = \text{eig}(D^T D) \]
- SVD factorizes the data matrix \( D \) as:
\[ D = UC \]
\[ B = U \Sigma V^T \]
\[ D = U \Sigma V^T \]

PCA

Assuming 0 mean data, the basis \( B \) that preserve the maximum variation of the signal is given by eigenvectors \( DD^T \).

\[ DD^T B = BA \]
Error function for PCA

- PCA minimizes the reconstruction error.
  
  \[ E(B, C) = \sum_{i=1}^{n} \| d_i - Bc_i \|_2 = \| D - BC \|_F \]

- Not unique solution: \( B \equiv R^{-1} C \in \mathbb{R}^{k \times c} \)
- To obtain same PCA solution \( R \) has to satisfy:
  \[
  \begin{align*}
  \hat{B} &= BR \\
  \hat{C} &= R^{-1} C \\
  \hat{B}^T \hat{B} &= I \\
  \hat{C} \hat{C}^T &= \Lambda
  \end{align*}
  \]

- \( R \) is computed as a generalized \( k \times k \) eigenvalue problem.
  \[ (CC^T)^{-1} R = B^T B \Lambda R^{-1} \]  
  (de la Torre, 2006)

PCA/SVD in computer vision

- PCA/SVD has been applied to:
  - Recognition (eigenfaces: Turk & Pentland, 1991; Sirovich & Kirby, 1987; Leonardis & Bischof, 2000; Gong et al., 2000; McKenna et al., 1997a)
  - Parameterized motion models (Yacoob & Black, 1999; Black et al., 2000; Black, 1999; Black & Jepson, 1996)
  - Appearance/shape models (Cootes & Taylor, 2001; Cootes et al., 1998; Pentland et al., 1994; Jones & Poggio, 1998; Casia & Solaroff, 1999; Black & Jepson, 1996; Blanz & Vetter, 1995; Cootes et al., 1995; McKenna et al., 1997; de la Torre et al., 1998b; de la Torre et al., 1998b)
  - Dynamic appearance models (Soatto et al., 2001; Rao, 1997; Orriols & Bireta, 2001; Gong et al., 2000)
  - Structure from Motion (Tomasi & Kanade, 1992; Bregler et al., 2000; Sturm & Triggs, 1996; Brand, 2001)
  - Illumination based reconstruction (Hayakawa, 1994)
  - Visual servoing (Murase & Nayar, 1995; Murase & Nayar, 1994)
  - Visual correspondence (Zhang et al., 1995; Jones & Malik, 1992)
  - Camera motion estimation (Hartley, 1992; Hartley & Zisserman, 2000)

More PCA/SVD work

- PCA/SVD has been applied to:
  - Image watermarking (Liu & Tan, 2000)
  - Signal processing (Mooren & de Moor, 1995)
  - Neural approaches (Oja, 1982; Sanger, 1989; Xu, 1993)
  - Bilinear models (Tenenbaum & Freeman, 2000; Marinont & Wandell, 1992)
  - Direct extensions (Welling et al., 2003; Penev & Atick, 1996)
- And many more (google)...
  - Results 1 - 10 of about 1,870,000 for "principal component analysis".
- Results 1 - 10 of about 65,300,000 for "Britney spears".

1-Robust PCA

- Two types of outliers:
  - Sample outliers (Xu & Yullie, 1995)
  - Intra-sample outliers (de la Torre & Black, 2001b; Skočaj & Leonardis, 2003)
- Standard PCA solution (noisy data):
Robust PCA

- Using robust statistics:
  (de la Torre & Black, 2001b; de la Torre & Black, 2003a)

\[ E_{rPCA}(B, C, \mu) = \sum_{i=1}^{d} \sum_{p=1}^{d} \rho(d_{ip} - (\mu_p + \sum_{j=1}^{k} b_{jp} c_{jp})) \]

First eigenvector with highest eigenvalue.

\[ A' = A - \lambda_1 u_1 u_1^T \]

Second eigenvector with highest eigenvalue.

\[ A'' = A' - \lambda_2 u_2 u_2^T \]

- In the robust case all the basis have to be computed simultaneously (including the mean).

Numerical problems

- No closed form solution in terms of an eigen-equation.
- Deflation approaches do not hold.

How to optimize it?

\[ E_{rPCA}(B, C, \mu) = \sum_{i=1}^{d} \sum_{j=1}^{d} \rho(d_{ij} - \mu_j - \sum_{k=1}^{k} b_{jk} c_{jk}) \]

- Normalized Gradient descent
  \[ B_{n+1} = B_n - H_b \]
  \[ H_b = \max_{b} \text{diag} \left( \frac{\partial^2 E_{rPCA}}{\partial b \partial b} \right) \]

- Deterministic annealing methods to avoid local minima.
  (Blake & Zisserman, 1987)

Example

- Small region
- Short amount of time

Statistical outlier

ORIGINAL  PCA  RPCA
Robust PCA

Original PCA RPCA Outliers

Component Analysis for Computer Vision F. De la Torre ECCV-06 17

Structure from Motion

Original Data SVD

Projection Outliers Robust SVD

Component Analysis for Computer Vision F. De la Torre ECCV-06 18

Related RPCA work

- Robust estimation of coefficients
- Robust estimation of basis and coefficients
  (Gabriel & Odoro, 1984; Croux & Filzmoser, 1981; Skocaj et al., 2002; Skocaj & Leonardis, 2003; de la Torre & Black, 2001b; de la Torre & Black, 2003a)
- Other Robust PCA techniques (sample outliers)
  (Campbell, 1980; Ruymagaart, 1981; Xu & Yulke, 1995)

Component Analysis for Computer Vision F. De la Torre ECCV-06 19

2- PCA with uncertainty and missing data

- Adding uncertainty
  \[ E_j(B,C) = \|W_i D - B C\|_1 = \sum_{ij} w_{ij}(d_{ij} - \sum_k h_{ij})^2 \]

- If weights are separable \( w = w_i w_j \) close form solution.
  \[ w = \begin{pmatrix} w_1 & \cdots & w_n \\ w_1' & \cdots & w_n' \end{pmatrix} \]
  \[ W \in \mathbb{R}^{n \times n} \]
  \[ w_i, w_j \geq 0 \]
  \( \text{Hadamard product} \)

- Generalized SVD
  (Greenacre, 1984; Irani & Anandan, 2000;)

Component Analysis for Computer Vision F. De la Torre ECCV-06 20
General case

- For arbitrary weights no closed-form solution.
  \[ E_2 (B, C) = \|W(D - BC)\| = \sum_{i=1}^{n} d_i (d_i - C)^T \text{diag}(w_i) (d_i - C) \]

- Alternated least squares algorithms
  - Slow convergence, easy implementation.
  - Damped Newton Algorithm
    - Fast convergence.
    - \( H \) definite positive: \( \frac{\partial^2 E_2}{\partial v^2} + \lambda I \)

Experiments

- 240 x 1027, 70% known
- Total: 500 runs

Related work

- Iterative (Wiberg, 1976; Shum et al., 1995; Morris & Kanade, 1998; Aans et al., 2002; Guerrero & Aguir, 2002)
- Closed-form (Aguir & Moura, 1999; Irani & Anandan, 2000)
- Power factorization (Hartley & Schaalitzky, 2003)
- Bayesian estimation (L.Torresani & Bregler, 2004)

Incremental PCA

- (de la Torre et al., 1998b; Ross et al., 2004; Brand, 2002; Skocaj & Leonardis, 2003; Champagne & Liu., 1998; A. Levy, 2000)

3- Parameterized Component Analysis (PaCA) (de la Torre & Black, 2003b)

- Learn a subspace invariant to geometric transformations?
- Data has to be **geometrically** normalized
  - Tedious manual cropping.
  - Inaccuracies due to matching ambiguities.
  - Hard to achieve sub-pixel accuracy.
Error function for PaCA

\[ E(B, C, a) = \sum_{i=1}^{T} \left[ \sum_{l=1}^{L} \left( d_i(x_{a_l}) - B e_t \right)^2 + \lambda_1 a_t + \lambda_2 c_t + \lambda_3 (a_t - B e_t)^2 \right] \]

Motion (warping) \hspace{1cm} Basis (B) & coefficients (c) \hspace{1cm} Regularization

Solving the optimization problem

- Linearizing the motion (Bergen et al., 1992; Black & Jepson, 1998)
  \[ d_i(f(x, a_i)) = d_i(f(x, a_{0i})) + J_i \Delta a_i \]
- Normalized gradient descent w.r.t. all parameters + deterministic annealing.
  - Update for c (appearance) & a (motion).
  - Updated for B (appearance basis).
- It is a non-convex function.
  - Stochastic initialization (G.A). (Lanitis et al., 1995; de la Torre & Black, 2000b)
  - Multiresolution motion estimation framework.

EigenEye Learning

More on parameterized CA

- Probabilistic model
  - Search scales exponentially with the number of motion parameters (Frey & Jojic, 1999a; Frey & Jojic, 1999b; Williams & Titsias, 2004)
- Other continuous approaches.
  (Schewitzer, 1999; Rao, 1999; Shashua et al., 2002)
- Invariant clustering
  (Fitzgibbon & Zisserman, 2003)
- Non-rigid motion
  (Baker et al., 2004)
- Invariant recognition
  (Black & Jepson, 1998)
- Invariant support vector machines (Avidan, 2001)
Bias solution

3 principal components (38 people)

Original

Bias

Unbiased

4- PCA over continuous spaces

(Levin & Shashua, 2002)

• PCA assumes discrete samples, but in the limit:
\[
\frac{1}{N} \sum_{i=1}^{N} d_i d_i^T \rightarrow \int f(d) d d^T d d
\]

• Sometimes not uniform sampling of \( f(d) \).

3 principal components (38 people)

Bias solution

• 3 principal components (38 people)

Weighting the data.

– Not clear which is the optimal weight. \( E(B,C) = \sum_i |B_i - B C_i| \).

More elegant approach.

– Find the principal components over the data points represented by (dense) uniform sampling.

– Integrating over the convex combination of the examples is less sensitive to the particular sample

\[
\sum_{i=1}^{n} w_i \lambda_{ii} \rightarrow \| \mathbf{W} \|
\]

How to unbiased the solution?

• Weighting the data.
  – Not clear which is the optimal weight. \( E(B,C) = \sum_i |B_i - B C_i| \).

• More elegant approach.
  – Find the principal components over the data points represented by (dense) uniform sampling.
  – Integrating over the convex combination of the examples is less sensitive to the particular sample

PCA OVER POLYTOPS

• PCA over polytops

\[
\text{Cov}(W) = \frac{1}{V(W)} \int W(W)\text{Cov}(W) dW
\]

Volume of polytops

- A simple 2D case, two points \( a_1 \) and \( a_2 \):
  - Regular PCA \( a_1 a_1^T + a_2 a_2^T \)
  - Integrating over a line \( \lambda a_1 + (1 - \lambda) a_2 \), \( 0 \leq \lambda \leq 1 \)

\[
\text{bias} = \int_{\lambda=0}^{1} \| a \|^2 d\lambda = \int_{\lambda=0}^{1} \text{Tr}(a^T a) d\lambda = \frac{1}{2} \int_{\lambda=0}^{1} |\lambda(1-\lambda)| d\lambda = \frac{1}{6}
\]

\[
\text{bias} = \min_{|\lambda|=1} \| a \|^2 = \min_{|\lambda|=1} \text{Tr}(a^T a) = \min_{|\lambda|=1} \text{Tr}(A^T A)
\]

- \( A = \begin{bmatrix} 1 & 0.5 & 0.5 \end{bmatrix} \)

\[
A^T = A A^T
\]
In general

\[
\Phi_i = \frac{1}{k(k+1)} \begin{bmatrix} 2 & 1 & \cdots & 1 \\ 1 & 2 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 2 \end{bmatrix} = \frac{1}{k(k+1)} (I + ee^T)
\]

\[
Cov(W) = \frac{1}{V(W)} \int_{acW} dd^Tdd = \frac{1}{k(k+1)} D(I + ee^T)D^T
\]

- Traditional PCA: \( \frac{1}{k} DD^T \)
- Continuous PCA: \( \frac{1}{k (k + 1)} DD^T + ee^T DD^T \)

5-Filtered PCA
(Bischof et al., 2004; Wildenauer et al., 2002)

- How to construct eigenspaces robust:
  - Varying illumination.
  - Occlusion.
  - Noise.
- Filtered PCA.
  - \( H \) is a convolution matrix (block circulant structure)
  - The coefficients of \( c_i \) remain the same under a convolution.

\[
H \in \mathbb{R}^{d \times d} \quad d_i \in \mathbb{R}^d \quad \Rightarrow \quad H d_i \approx (HB)c_i
\]

Filtered PCA

- We can filter the image with many filters:

  - Eigenbasis
  - Steerable filters (Simoncelli et al., 1992)

\[
H' d_i = (H'B)c_i
\]

- Filtered Eigenbasis
- Given a new image, compute the filtered representation and robustly compute \( c_i \) (eigenbasis known).

Experiments of Filtered PCA
More results

<table>
<thead>
<tr>
<th>Obj.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robust method - all eigenvectors used</td>
<td>1</td>
<td>360</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100.0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>323</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>99.8</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>503</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>99.8</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>505</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>98.8</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>62</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100.0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>11</td>
<td>62</td>
<td>12</td>
<td>1</td>
<td>8</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>14</td>
<td>0</td>
<td>8</td>
<td>22</td>
<td>10</td>
<td>413</td>
<td>24</td>
<td>7</td>
<td>4</td>
<td>99.9</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>51</td>
<td>37</td>
<td>1</td>
<td>26</td>
<td>42</td>
<td>6</td>
<td>65.7</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>16</td>
<td>2</td>
<td>9</td>
<td>17</td>
<td>3</td>
<td>7</td>
<td>439</td>
<td>1</td>
<td>87.3</td>
<td>14.60</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>9</td>
<td>46</td>
<td>0</td>
<td>5</td>
<td>39</td>
<td>599</td>
<td>5.34</td>
</tr>
<tr>
<td>Avg.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>96.5</td>
</tr>
</tbody>
</table>

Standard method - all eigenvectors used

<table>
<thead>
<tr>
<th>Obj.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robust method - all eigenvectors used</td>
<td>1</td>
<td>74</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>40</td>
<td>179</td>
<td>41</td>
<td>4</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>215</td>
<td>0</td>
<td>6</td>
<td>3</td>
<td>67</td>
<td>0</td>
<td>1</td>
<td>20</td>
<td>15</td>
<td>66.4</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>7</td>
<td>237</td>
<td>3</td>
<td>101</td>
<td>28</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>35</td>
<td>47.0</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>296</td>
<td>43</td>
<td>28</td>
<td>0</td>
<td>9</td>
<td>44</td>
<td>4</td>
<td>65.8</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>553</td>
<td>26</td>
<td>7</td>
<td>17</td>
<td>19</td>
<td>0</td>
<td>87.4</td>
<td>9.73</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>78</td>
<td>0</td>
<td>0</td>
<td>67</td>
<td>553</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>14</td>
<td>76.8</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>19</td>
<td>0</td>
<td>20</td>
<td>96</td>
<td>74</td>
<td>101</td>
<td>44</td>
<td>22</td>
<td>48</td>
<td>35.9</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>30</td>
<td>0</td>
<td>0</td>
<td>134</td>
<td>124</td>
<td>3</td>
<td>10</td>
<td>22</td>
<td>0</td>
<td>27.5</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>54</td>
<td>82</td>
<td>102</td>
<td>4</td>
<td>31</td>
<td>255</td>
<td>6</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>10</td>
<td>2</td>
<td>24</td>
<td>63</td>
<td>175</td>
<td>5</td>
<td>4</td>
<td>36</td>
<td>267</td>
<td>58.4</td>
</tr>
<tr>
<td>Avg.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>74.3</td>
</tr>
</tbody>
</table>

6- PCA of a Set of Rotated Images

(Uenohara & Kanade, 1998; Jogan et al., 2003)

- Set of uniformly in-plane rotated versions of the same object (0 background).

- \( \mathbf{D}^T \mathbf{D} \) is a circulant symmetric Toeplitz matrix.

\[
\mathbf{D}^T \mathbf{D} = \begin{bmatrix}
  d_1^2 & d_1 d_2^* & \ldots & \ldots & \ldots & \ldots & d_1 d_n^* \\
  d_1 d_2^* & d_2^2 & \ldots & \ldots & \ldots & \ldots & \ldots \\
  \ldots & \ldots & \ldots & d_2 d_{n-1}^* & \ldots & \ldots & \ldots \\
  \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
  \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
  \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
  d_n^* d_1 & \ldots & \ldots & \ldots & d_n^* & d_n^2 \\
\end{bmatrix}
\]

Eigenfiltering for flexible Eigentracking

(de la Torre et al., 2000)

\[
\sum_{i=1}^{N} \frac{1}{k} (\mathbf{I}_F (x_i) - \mathbf{m}_F) \mathbf{v}_i^T (\mathbf{I}_F (x_i) - \mathbf{m}_F) 
\]

\[
\mathbf{x}_i = \mathbf{f} (\mathbf{x}_i, \mathbf{a}) + \sum_{k=1}^{K} b_k (\mathbf{u}_k, \mathbf{a})
\]

Rigid parameters

Non-rigid parameters

\[
\mathbf{I}_F (x_i) = \begin{bmatrix}
  \mathbf{H}'_d' \\
  \ldots \\
  \mathbf{H}'d \\
\end{bmatrix}
\]

Eigenvectors of circulant matrices

- The eigenvectors of circulant matrices are the n basis of the Fourier matrix.

\[
\mathbf{F} = \begin{bmatrix}
  1 & 1 & \ldots & 1 \\
  1 & 1 & \ldots & 1 \\
  \ldots & \ldots & \ldots & \ldots \\
  1 & 1 & \ldots & 1 \\
\end{bmatrix}
\]

\[
\mathbf{V} = \begin{bmatrix}
  \sum_{l=1}^{N} d_l e^{2 \pi \imath l k} \\
  \sum_{l=1}^{N} d_l e^{2 \pi \imath l k} \\
  \sum_{l=1}^{N} d_l e^{2 \pi \imath l k} \\
  \ldots \\
  \sum_{l=1}^{N} d_l e^{2 \pi \imath l k} \\
\end{bmatrix}
\]

\[
\mathbf{F} = \mathbf{V} \cdot \mathbf{D}
\]

- Circulant and SYMMETRIC \( d_i = d_{N-i} \). \( \mathbf{F} \) is the basis of cosines.

\[
eig(\mathbf{DD}^T) \quad \rightarrow \quad \text{DCT}
\]
Generalization to multiple templates
(Jogan et al., 2003)

- $P$ different locations (objects), each shifted $N$ times.
- every $Q_{ij}$ is circulant (but in general not symmetric!)

$A = B' D = \begin{bmatrix}
Q_{0} & \cdots & \cdots & \cdots & \cdots \\
Q_{1} & & & & \\
\vdots & & \ddots & & \\
Q_{N-1} & & & Q_{0}
\end{bmatrix}$

Same eigenvectors

- Eigencalculation of $O(N^3)$ rather than $O(NP^3)$

7- Mixture of subspaces
(Vidal et al., 2003)

- How to estimate (basis) of a mixture of subspaces and varying dimensions?
  - Multibody structure from motion
    - 2D & 3D Motion segmentation.

Example: clustering in 1D

$x^2 - (b_1 + b_2)x + b_1 b_2 = 0$

$x = b_1 \text{ or } x = b_2$

$(x - b_1)(x - b_2) = 0$

Number of groups?

$\text{rank}(P) = 1$: one group only
$\text{rank}(P) = 2$: two groups
### Example 2

How to compute $n$, $c$, $b$’s?

- Number of clusters
  
  $$n = \min \{ i : \text{rank}(P_i) = i \}$$

- Cluster centers
  
  Roots of $p_n(x)$

- Solution is unique if
  
  $N_{\text{points}} \geq n_{\text{groups}}$

- Solution is closed form if
  
  $n_{\text{groups}} \leq 4$

\[ p_n(x) = x^n + c_1 x^{n-1} + \cdots + c_n = 0 \]
\[ p_n(x) = \begin{bmatrix} x^n & \cdots & x_1 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ x_1 & \cdots & x_N & 1 \end{bmatrix} c = 0 \]

### GPCA

- One plane
  
  $b_i^T x = b_1 x_1 + b_2 x_2 + b_3 x_3 = 0$

- One line
  
  $b_i^T x = b_1 x_1 + b_2 x_2 + b_3 x_3 = 0$
  
  $b_i^T x = b_4 x_1 + b_5 x_2 + b_6 x_3 = 0$

- One subspace can be represented with
  
  Set of linear equations $S = \{ x : b_i^T x = 0 \}$

- Set of polynomials of degree 1

### Examples

- Black group
- Gray group
- White group

De Morgan’s rule

$S_1 \cup S_2 = \{ x \mid (b_i^T x)(b_j^T x) = 0 \}$ and $(b_i^T x)(b_j^T x) = 0$

A union of $n$ subspaces can be represented with a set of homogeneous polynomials of degree $n$
Multiple eigenspaces

(Leonardis et al., 2002)

• Goal: group visually similar images into categories in an
unsupervised and self-organising way.

Multiple eigenspaces – idea

• Group the images and construct multiple eigenspaces
such that:
  – Mean reconstruction error is always below a threshold
  – The dimensionalities of eigenspaces are as small as possible
• Start from a seed and then grow hypotheses
• Grow and select paradigm:
  – Simultaneously and independently grow multiple competing
    hypotheses
  – Select the best hypotheses at the end

Multiple eigenspace - example

• 5 categories, 21
images from each

Growing:

Results:
• mean eigenvectors corresponding images
  ES1:
  ES2:
  ES3:
  ES4:
  ES5:
“Intercorrelations among variables are the bane of the multivariate researcher’s struggle for meaning”

Cooley and Lohnes, 1971

Part-based representation

- The firing rates of neurons are never negative
- Independent representations.

NMF & ICA

Non-negative Matrix Factorization

- Positive factorization.
  \[ E(B, C) = \| D - BC \|_F \quad B, C \geq 0 \]
- Leads to part-based representation.

Nonnegative factorization algorithm

(See Lee & Seung, 1999; Lee & Seung, 2000)

\[
\min_{W_0, V_0} F = \sum_y \| d_y - (BC)_y \|^2
\]

Inference:

\[
C_y \leftarrow C_y \frac{(B^T D)_y}{(B^T B)_y}
\]

Learning:

\[
B_y \leftarrow B_y \frac{(D C)_y}{(B C C^T)_y}
\]

- Multiplicative algorithm can be interpreted as diagonally rescaled gradient descent.
8- PCA & NMF for clustering

(Zha et al., 2001; Ding & He, 2004; de la Torre & Kanade, 2006)

- VQ, k-means
  \[ E(G, B) = \| D - BG \|_F = \sum_{i=1}^{n} \sum_{j \in C_i} \| d_i - b_j \| \]

  \[ G' = \begin{bmatrix} 1 & \ldots & 0 \\ 0 & \ldots & 1 \\ 0 & \ldots & 0 \end{bmatrix} g_y \in \{0,1\} \quad G1_{c} = 1 \quad G \in \mathbb{R}^{n \times 2} \]

- After eliminating \((B)\)
  \[ E(G) = \| D - DG(G^2G)^{-1}G' \|_F = tr(D^T D) - tr((G^2G)^{-1}G^2D')DG) \geq \min_{i,j} \lambda_j \]

- Relaxing binary constraints
  \[ E(G) = tr((G^2G)^{-1}G' G^T D' D G) \]

- Eigenvectors (PCA) of \(D^TD\) are the optimal continuous solution of the indicator variable \(G\).

---

Examples

- Soft clustering

  \[ R_{ij}=1 \text{ if } d_i \text{ and } d_j \text{ are cluster together.} \]

  \[ R_{ij}=-1 \text{ if } d_i \text{ and } d_j \text{ are not cluster together.} \]

- Adding side-information

  \[ K' = K + \alpha \Gamma \] (Zass & Shashua, 2005)

---

Clusterung with NMF

(Zass & Shashua, 2005; Ding et al., 2005)

\[ E(G, B) = \| \Gamma - BG \|_F \quad \Gamma = [\varphi(d_1) \varphi(d_2) \ldots \varphi(d_n)] \]

- Soft clustering and non-negative matrix factorization:

  \[ \Gamma^T \Gamma - GG^T \|_F \geq 0 \quad F = G^T G = I \quad F_1 = 1 \quad F'1 = 1 \]

  Affinity matrix

  \[ G = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{\nu_1} \\ \vdots & \vdots \\ 0 & \sqrt{\nu_n} \end{bmatrix} \]

- Previous normalization \( \min_K \| \Gamma^T \Gamma - K \|_F \) s.t. \( K1 = K^T1 = \beta1 \)

- Gradient descent with multiplicative updates.

---

Independent Component Analysis

- We need more than second order statistics to represent the signal.
ICA & Signals

Source A

Mixture 1

ICA

Source A

Mixture 2

ICA & Signals

ICA

\( \text{ICA} \)

\( \text{(Hyvärinen et al., 2001)} \)

\[ D = BC \quad C = S = WD \quad W \approx B^{-1} \]

- Look for \( s_i \) that are independent.
- PCA finds uncorrelated variables, the independent components have non-Gaussian distributions.
- Uncorrelated \( E(s_i s_j) = E(s_i)E(s_j) \)
- Independent \( E(g(s_i)f(s_j)) = E(g(s_i))E(f(s_j)) \) for any non-linear \( f, g \)

ICA vs PCA

ICA vs PCA

ICA vs PCA

Many optimization criteria

- Minimize high-order moments: e.g. kurtosis
  \[ \text{kurt}(W) = E(s^4) - 3(E(s^2))^2 \]
- Many other information criteria.
- Also an error function: (Olhausen & Field, 1996)
  \[ \sum_{i=1}^{n} \| H_i - Bc \| + \sum_{i=1}^{n} \delta(C_i) \]
  Sparseness (e.g. \( S=1 \))
- Other sparse PCA.
  (Chennubhotla & Jepson, 2001b; Zou et al., 2005; dAspremont et al., 2004;)

ICA vs PCA

ICA vs PCA

ICA vs PCA

ICA vs PCA
Basis of natural images

Denoising

Original image
Noisy Image (30% noise)
Denoise (Wiener filter)
ICA

Linear Discriminant Analysis (LDA)
(Fisher, 1936; Mardia et al., 1979; Bishop, 1995)

$S_b = \sum_{i=1}^{C} \sum_{j=1}^{C} (\mu_i - \mu_j)(\mu_i - \mu_j)^T$

$S_w = \sum_{i=1}^{C} \sum_{j=1}^{C} (d_i - \mu_j)(d_i - \mu_j)^T$

$S_b B = S_w B A$

$S_w = \sum_{i=1}^{C} \sum_{j=1}^{C} (d_i - \mu_j)(d_i - \mu_j)^T$

- Optimal linear dimensionality reduction if classes are Gaussian with equal covariance matrix.

Discriminative Models

- Linear Discriminant Analysis (LDA)
  9) Multimodal Oriented Discriminant Analysis.
  10) Discriminative Cluster Analysis.
  11) Robust Linear Discriminant Analysis.
- Oriented Component Analysis (OCA)
  12) Representational Oriented Component Analysis.
- Canonical Correlation Analysis (CCA)
  13) Dynamical Coupled Component Analysis.
  14) CCA and Mobile robotics applications.
- Relevance Component Analysis (RCA)
Matrix formulation

where 
\[ S_t = \frac{1}{n} \sum_{i=1}^{n} (d_i - m)(d_i - m)^T = \frac{1}{n-1} \mathbf{DP}_1 \mathbf{D}^T \]
\[ S_w = \frac{1}{n} \sum_{i=1}^{n} \sum_{d_j \in C_i} (d_j - m_i)(d_j - m_i)^T = \frac{1}{n-1} \mathbf{DP}_2 \mathbf{D}^T \]
\[ S_b = \sum_{i=1}^{n} \frac{n_i}{n-1} (m_i - m)(m_i - m)^T = \frac{1}{n-1} \mathbf{DP}_3 \mathbf{D}^T \]

- \( m \) overall mean, \( m_i \) mean for class i.
- \( \mathbf{P}_i \) is a projection matrix (\( \mathbf{p}_i = \mathbf{p}_i^T \))

Error function for LDA

\[ E(\mathbf{V}, \mathbf{B}) = \parallel (\mathbf{G}^T \mathbf{G})^{\frac{1}{2}} (\mathbf{G}^T - (\mathbf{B}^T \mathbf{D})) \parallel \]

Where are linear methods applicable?

\[ \mathbf{S}_B \mathbf{V} = \mathbf{S}_W \mathbf{V} \Lambda. \]

When is LDA good?

- The discriminant power of the generalized eigenvalue decomposition equation
  \[ \mathbf{M}_A \mathbf{V} = \mathbf{M}_B \mathbf{V} \Lambda \]
  is \( \text{tr}(\mathbf{M}_B^{-1} \mathbf{M}_A) \), which is the same as
  \[ K = \sum_{i=1}^{n} \sum_{j=1}^{p} \frac{\lambda_{i+}}{\lambda_{B_{ij}}} (b_j a_i)^2. \]
- Large values of \( K \) indicates instability in the results.
- If for some \( (i,j) \) max \( (b_j a_i)^2 \) close to 1 the solution might be unstable.
9. Multimodal Oriented Component Analysis (MODA)  
(de la Torre & Kanade, 2005a)

- How to extend LDA
  - Model class covariances.
  - Multimodal classes.
  - Deal efficiently with huge covariance matrices (e.g. 100*100).

Multimodality

![LDA vs MODA](image)

Multimodal Oriented Discriminant Analysis

- \( B \) that MAXIMIZES the Kullback-Leibler divergence between clusters among classes.

\[
\sum_{i=1}^{C} \sum_{r} \sum_{i \neq r} \sum_{c \neq i} \sum_{c \neq r} (B^T \Sigma_i^{-1} + \Sigma_r^{-1} + \Sigma_i^{-1} + \Sigma_r^{-1})(\mu_i - \mu_r)^T (\mu_i - \mu_r) \text{ (B)}
\]

- 1 mode per class and equal covariances equivalent to LDA.

Optimization

- Hard optimization problem
  \[
  J(B) = -\sum_{i=1}^{C} tr((B^T \Sigma_i^{-1} + (B^T A B))
  \]

- Iterative Majorization  
  \( T(B) \geq J(B) \forall B 
  T(B_0) = J(B_0) \)

\( W_0 \)

\( W_1 \)
Component Analysis for Computer Vision  
F. De la Torre  
ECCV-06

Majorization

\[
T(B) = \sum_{i=1}^{classes} \left\| (B^T \Sigma_i B)^{-1} B^T A_i - (B^T \Sigma_i B)^{-1} (B^T \Sigma_i B_0) B^T A_i \right\|
\]

\[
\geq - \sum_{i=1}^{classes} \text{tr}((B^T \Sigma_i B)^{-1} (B^T A_i B))
\]

- Slow convergence, first gradient descent:

\[
B^{(n+1)} = B^{(n)} - \eta \frac{\partial E_i(B^{(n)})}{\partial B}
\]

\[
\frac{\partial E_i}{\partial B} = \sum_{i=1}^{classes} A_i B^{(n)} (B^{(n)}^T \Sigma_i B^{(n)} - A_i^T \Sigma_i A_i)^{-1} - \sum_{i=1}^{classes} \left( B^{(n)} (B^{(n)}^T \Sigma_i B^{(n)} - A_i^T \Sigma_i A_i)^{-1} (B^{(n)}^T A_i B^{(n)} - B^{(n)}^T A_i B_0)ight)
\]

Face recognition from video

- Challenges
  - Low quality small images (40-50 pixels).
  - Changes in expression/pose/occlusion/illumination.
  - Real time and scalable to several users.

Prototypes

Adding space-time constraints

(de la Torre et al., 2005b)

Number of basis

Recognition

LDA  MODA  PCA

Prototypes

40*40 16

B

(modification of de la Torre et al., 2005b)
### 10- Robust Linear Discriminant Analysis

(Fidler et al., 2006)

- How to construct robust discriminative models?

No noise model.

\[ G = B^T D \]

- Remove outliers in the space of generative models and learn linear a mapping to discriminative models.
- Assume \( (d >> n) \) and all the samples independent.

\[
\begin{align*}
G &= B^T D & \text{discriminative} \\
B^d &= \{B^e \} & \text{generative}
\end{align*}
\]

\[ G \in \mathbb{R}^{n \times c} \]

\[ B^d \in \mathbb{R}^{d \times c} \]

\[ D \in \mathbb{R}^{d \times n} \]

\[ C \in \mathbb{R}^{c \times c} \]

\[ G \in \mathbb{R}^{n \times c} \]

\[ D \in \mathbb{R}^{d \times n} \]

### Robust LDA

- \( B^d \) as linear combination of the basis of \( B^g \).

\[ B^d = B^d V = \begin{bmatrix} V_{1,k} \\ \vdots \\ V_{n-k} \end{bmatrix} \quad V \in \mathbb{R}^{n \times c} \]

\[ G = B^d T D = V^T B^g T D = V^T C_{1,k} + V^T C_{k-n} \]

- If just \( V^T C_{1,k} \) discriminative power can be lost.
- Add \( c \) basis to fully recover the LDA solution.

\[
\begin{align*}
W &= B^e V_{n-k,c} \in \mathbb{R}^{n \times c} \\
W &= W(W^T W)^{-1} \\
B^d &= \{B^e \} \\
V &= V_{1} \left( \frac{W}{W^T W} \right)^{1/2} \\
G &= \left( V \right)^T B^g T D
\end{align*}
\]

### Robust PCA

- Given training data find \( B^e, \quad V, \quad B^e = \{B^e \} \)
- Find robustly the coefficients in the \( B^e \) basis such that: \( d_i = B^e c \) and then estimate \( G \).

- Robust estimation is achieved by hypothesize-and-test paradigm.

\[
\begin{bmatrix} d_{i1} \\ \vdots \\ d_{id} \end{bmatrix} = \begin{bmatrix} b_{11} c_1 + \ldots + b_{1k} c_k \\ \vdots \\ b_{dk} c_d + \ldots + b_{dk} c_k \end{bmatrix}
\]

### Experimental results
Component Analysis for Computer Vision

More examples

<table>
<thead>
<tr>
<th>Gesture</th>
<th>Casual</th>
<th>Recognition rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>sun</td>
<td>0.23</td>
<td>43.00 93.00</td>
</tr>
<tr>
<td>scarf</td>
<td>0.46</td>
<td>42.00 72.00</td>
</tr>
<tr>
<td>screen</td>
<td>0.85</td>
<td>87.00 7.00</td>
</tr>
</tbody>
</table>

(De la Torre & Kanade, 2006)

11-Discriminative Cluster Analysis (DCA)

- Generative clustering (e.g. k-means):
\[ E(G, B) = \| D - BG^T \|_F = \sum_{i=1}^{n} \sum_{j \in C_i} \| d_j - b_i \| \]
\[ g_{ij} \in \{0,1\} \quad G 1_n = I_n \]
- Not efficient for high dimensional data.
- Multiple local minima.

- Discriminative clustering (De la Torre & Kanade, 2006):
\[ E(V, B, G) = \| (G^T G)^{-1/2} (G^T - VB^T D) \|_F \]
- Simultaneous dimensionality reduction and clustering.

Optimization

- Eliminate V
\[ E(B, G) \propto tr((B^T D B)^{-1} (B D G (G^T G)^{-1} G^T D B)) \]
- Optimize for B
\[ DG(G^T G)^{-1} G^T D B = DD^T B \lambda \]
- Optimize for G
\[ A = C^T (C^T C)^{-1} C \quad \lambda = B^T D \]
\[ G = V \phi \quad V^{(t+1)} = V^{(t)} - \eta \frac{\partial E}{\partial V} \]
\[ \frac{\partial E}{\partial V} = (I_n - G (G^T G)^{-1} G^T) AG (G^T G)^{-1} \]

Experiments
**Related LDA work**

- **Face recognition** (Belhumeur et al., 1997; Zhao, 2000; Martinez & Kak, 2003)
- **Small sample problem** (Chen et al., 2000; Yu & Yang, 2001)
- **Mixture** (Hastie et al., 1995; Zhu & Martinez, 2006)
- **Neural approaches** (Gallinari et al., 1991; Lowe & Webb, 1991)
- **Heteroscedastic Discriminant Analysis** (Kumar & Andreou, 1998; Fukunaga, 1990; Mardia et al., 1979; Saon et al., 2000)

**Oriented Component Analysis (OCA)**

- Generalized eigenvalue problem: $\Sigma_f b_k = \Sigma_c b_k \lambda$
- $b_{OCA}$ is steered by the distribution of noise.
12- Representational Oriented Component Analysis (ROCA)

**Challenges**
- Just 1 training image.
- Changes in appearance, expression and illumination.

**Interpretation of OCA**

\[
\Sigma_e = \Lambda U^T + \sigma^2 I
\]

\[
b_k = \sum_{e}^{-1} d_k
\]

\[
b_e = \Sigma^{-1} d_e \propto (I - U \left( \frac{\lambda - \sigma^2}{\lambda} \right) U^T) d_e
\]
Combining several representations

- Morphological filters, Gabor filters, edge detectors, derivatives of Gaussians, etc. (150 images)

$$\max B_k \left[ B_k^T \left( \sum_i \left( \frac{1}{\sigma^2} + \frac{1}{\sigma^2} \right) \right) B_i \right]$$

Solving Generalized Eigenvalue

- Rank deficient matrices.
- Not numerically stable algorithms/bad generalization (over fitting)
- Modified Subspace Iteration.

$$C\hat{V}_{k+1} = AV_k \quad \hat{V}_{k+1} = \frac{\hat{V}_{k+1}}{\max(\hat{V}_{k+1})}$$

$$S = V_{k+1}^T A V_{k+1} \quad T = V_{k+1}^T C V_{k+1}$$

$$SW = CW\Delta$$

$$\hat{V}_{k+1} = \hat{V}_{k+1} W$$

Combining Classifiers

- 150 Representations
- Combining classifiers
  * Normalization
  * Sum.

Results

- PCA –
  - Mah 10%
  - Euclidean 16%
- NN-- 29%
- Face it-- 40.9%
- ROCA
  - Max of individual classifiers 47%
  - Combining 62.1%

152 Subjects
13- Dynamic Coupled Component Analysis (DCCA) (de la Torre & Black, 2001a)

- Learning the coupling.
- High dimensional data.
- Limited training data.

Solutions?

- PCA independently and general mapping
  
- Signals dependent signals with small energy can be lost.

Dynamic Coupled Component Analysis

\[
E_{acc}(\mathbf{B}, \hat{\mathbf{B}}, \mathbf{A}, \mathbf{C}, \mu, \hat{\mu}) = \sum_{i=1}^{n} \| \mathbf{d}_i - \hat{\mathbf{B}} \mathbf{e}_i \|_{W_1}^2 \\
+ \lambda_1 \sum_{i=1}^{n} \| \mathbf{e}_i - \mathbf{B}^T (\mathbf{d}_i - \mu \mathbf{c}_i) \|_{W_2}^2 \\
+ \lambda_2 \sum_{i=1}^{n} \| \mathbf{e}_i - \mathbf{A} \mathbf{c}_i \|_{W_3}^2
\]

Projection

Dynamics
Robot localization with Canonical Correlation Analysis
(Skocaj & Leonardis, 2000)

![Diagram of robot localization with Canonical Correlation Analysis](image)

**Canonical Correlation Analysis (CCA)**
(Mardia et al., 1979; Borga)

- Learn relations between multiple data sets? (e.g. find features in one set related to another data set)
- Given two sets \( X \in \mathbb{R}^{d \times n} \) and \( Y \in \mathbb{R}^{d \times n} \), CCA finds the pair of directions \( w_x \) and \( w_y \) that maximize the correlation between the projections (assume zero mean data)
  \[
  \rho = \frac{w_x^T X Y w_y}{\|w_x\| \|w_y\|}
  \]
- Several ways of optimizing it:
  \[
  A = \begin{bmatrix} 0 & X^T Y \\ X^T Y & 0 \end{bmatrix} \in \mathbb{R}^{(d_1+d_2)(d_1+d_2)}, \quad B = \begin{bmatrix} X^T X & 0 \\ 0 & Y^T Y \end{bmatrix} \in \mathbb{R}^{(d_1+d_2)(d_1+d_2)}
  \]
  \[
  w = \lambda w
  \]
- An stationary point of \( r \) is the solution to CCA.
  \[
  A w = \lambda B w
  \]

More on Canonical Correlation Analysis

- If \( d_1 \gg n \) using the kernel trick efficient ways of solving it.
- Maximum number of canonical correlation vectors \( \min(d_1, d_2) \)
- Learn a linear mapping between \( Y \) and the projection of \( X \) into canonical components.
  \[
  \min_F \| Y - F(W^T X) \|_F
  \]
  \[
  F = Y(W^T X)^+
  \]

Examples

- Training images
  ![Examples of training images](image)
Relevant Component Analysis

Adding side-information

Example

Standard extensions

- Latent Variable Models
- Tensor Factorization
  - 2D PCA/LDA.
  - Higher order extension.
- Kernel Methods

Factor Analysis

- A Gaussian distribution on the coefficients and noise is added to PCA → Factor Analysis. (Mardia et al., 1979)

\[
d = \mu + Bc + \eta
\]

\[
p(c) = N(c | 0, I) \quad p(d | c, B) = N(d | \mu + Bc, \Psi)
\]

\[
p(\eta) = N(\eta | 0, \Psi) \quad \Psi = \text{diag}(\eta_1, \eta_2, \ldots, \eta_n)
\]

\[
\text{cov}(d) = E((d - \mu)(d - \mu)^T) = BB^T + \Psi
\]

- Inference (Roweis & Ghahramani, 1999; Tipping & Bishop, 1999a)

\[
p(c, d) \quad \text{Jointly Gaussian}
\]

\[
p(c | d) = N(c | m, V)
\]

\[
m = B^T (BB^T + \Psi)^{-1} (d - \mu)
\]

\[
V = (I + B^T \Psi^{-1} B)^{-1}
\]

PCA reconstruction low error.
FA high reconstruction error (low likelihood).
### Component Analysis for Computer Vision

**Ppca**

- If $\Psi = E(\eta \eta^T) = d_\eta$ PPCA.
- If $\epsilon \rightarrow 0$ is equivalent to PCA: $E \rightarrow 0 \quad B^T(\epsilon B^T + \Psi)^{-1} = (B^T)^{-1} B^T$
- Probabilistic visual learning (Moghaddam & Pentland, 1997)

$$p(d) = \frac{1}{2\pi^d \sigma^d} \exp\left( -\frac{d^T \Sigma^{-1} d}{2\sigma^2} \right)$$

$$c_i = B^T d_i$$

---

**A least squares interpretation**

- Directly minimizing

$$E(B, \Lambda, \sigma) = \sum \| B \Lambda B^T - \sigma^2 I_d \|_F$$

$$B^T B = I_d \quad \Lambda \text{ is diagonal}$$

derives in the same solution as PPCA (de la Torre & Kanade, 2005b).

$$\sum d \times d \quad B B^T + \sigma^2 I$$

$$d (d + 1) \approx \frac{d^2}{2} \quad k(2d - k + 1) \approx kd$$

---

**More on PPCA**

- Extension to mixtures of Ppca (mixture of subspaces).
  (Tipping & Bishop, 1999b; Black et al., 1998; Jebara et al., 1998)
- Tracking (Yang et al., 1999; Yang et al., 2000a; Lee et al., 2005; de la Torre et al., 2000b)
- Recognition/Detection (Moghaddam et al., 2000; Shakhnarovich & Moghaddam, 2004; Everingham & Zisserman, 2006)
- PCA for the exponential family (collins et al., 2001)

---

**Tensor Decomposition**

- 2D PCA/LDA
- General tensor factorization.
2D PCA/SVD

- Vectorizing images do not preserve 2D properties and spatial properties are lost.
- Many ways of stacking images into matrices

```
Original Images 6x6x4 N=400
```

SVD versus 2D SVD

\[
E(L, R, (M_j)) = \sum_{i=1}^{\infty} \|D_i - LM_iR^T\|^2
\]

\[
D_i \in \mathcal{R}^{nrc} \quad L \in \mathcal{R}^{nc \times l_1} \quad R \in \mathcal{R}^{c \times l_2} \quad M_j \in \mathcal{R}^{l_1 \times l_2}
\]

\[
L' L = I \quad R'R = I
\]

- Compression ratio \( \frac{nc \cdot cl_1 \cdot nl_2}{nrc} \)
- Recognition \( \|D_j - D_i\| = \|L(M_j - M_i)R\| = \|M_j - M_i\| \frac{rc}{l_1 l_2} \)
- 2D SVD smaller computational cost (space & time)
- Same or better reconstruction (for same number of parameters)

Optimization

- No closed form solution.
- Alternate

\[
\begin{align*}
\text{Iterate until convergence} & : \quad \sum_{i=1}^{\infty} D_i R R^T D_i^T L = L \Lambda_1 \\
\text{Iterate until convergence} & : \quad \sum_{i=1}^{\infty} D_i L L^T D_i R = R \Lambda_2 \\
M_i = L_i D_i R
\end{align*}
\]

2D PCA/SVD

- SVD \( k=15 \), storage 160560
- 2DSVD \( l_1=l_2=15 \), storage 93060
Recognition

Eigenfaces

- Facial images (identity change)
- Eigenfaces bases vectors capture the variability in facial appearance (do not decouple pose, illumination, ...)

Tensor faces

(Vasilescu & Terzopoulos, 2002; Vasilescu & Terzopoulos, 2003)

Data Organization

- Linear/PCA: \textbf{Data Matrix}
  - \( R \text{ people} \times \text{images} \)
  - a matrix of image vectors
- Multilinear: \textbf{Data Tensor} \( \mathcal{D} \)
  - \( R \text{ people} \times \text{views} \times \text{illums} \times \text{express} \times \text{pixels} \)
  - N-dimensional matrix
  - 28 people, 45 images/person
  - 5 views, 3 illuminations, 3 expressions per person
N-Mode SVD Algorithm

\[ D = Z \times U_{\text{people}} \times U_{\text{views}} \times U_{\text{illums}} \times U_{\text{express}} \times U_{\text{pixels}} \]

Strategic Data Compression = Perceptual Quality

- TensorFaces data reduction in illumination space primarily degrades illumination effects (cast shadows, highlights)
- PCA has lower mean square error but higher perceptual error

Results

<table>
<thead>
<tr>
<th>Data Set - 16,875 images</th>
<th>Training Images - 2,700</th>
<th>Test Images:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- 75 people</td>
<td>- 75 people</td>
<td>- 75 people</td>
</tr>
<tr>
<td>- 15 viewpoints</td>
<td>- 6 viewpoints</td>
<td>- 9 viewpoints</td>
</tr>
<tr>
<td>- 15 illuminations</td>
<td>- 6 illuminations</td>
<td>- 9 illums</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Linear Models</th>
<th>Multilinear Models</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCA</td>
<td>83%</td>
<td>93%</td>
</tr>
<tr>
<td>ICA</td>
<td>89%</td>
<td>97%</td>
</tr>
<tr>
<td>TensorFaces</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Independent</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TensorFaces</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Related work

- **Tensor factorization** (O’Leary & Peleg, 1983; Shashua & Levin, 2001; Paatero & Tapper, 1994; Shashua & Hazan, 2005)
- **2D PCA** (Kong & Wang, 2005; Ding & Ye, 2006; Zhang et al., 2006; Zhang & Zhou, 2005; Yang et al., 2004b)
- **2D LDA** (Ye, 2005; Ye et al., 2005; Liu et al., 1993)

Kernel Methods

- Given in the tutorial.