Component Analysis Methods for Signal Processing

Fernando De la Torre
ftorre@cs.cmu.edu

Component Analysis for PR

- 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.
Component Analysis for PR

- **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.**

---

Component Analysis (ICA)

[Diagram of ICA]
Why Component Analysis for PR?

- Learn from high dimensional data and few samples.
  - Useful for dimensionality reduction.

- Easy to incorporate
  - Robustness to noise, missing data, outliers (de la Torre & Black, 2003a)
  - Invariance to geometric transformations (de la Torre & Black, 2003b; de la Torre & Nguyen, 2007)
  - Non-linearities (Kernel methods) (Scholkopf & Smola, 2002; Shawe-Taylor & Cristianini, 2004)
  - Probabilistic (latent variable models) (Everitt, 1984)
  - Exponential family PCA (Gordon, 2002; Collins et al. 01)

- Efficient methods $O(d \times n < n^2)$

Are CA Methods Popular/Useful/Used?

- About 20% of CVPR-06 papers use CA.

- Google:
  - Results 1 - 10 of about 1,870,000 for "principal component analysis".
  - Results 1 - 10 of about 506,000 for "independent component analysis".
  - Results 1 - 10 of about 273,000 for "linear discriminant analysis".
  - Results 1 - 10 of about 46,100 for "negative matrix factorization".
  - Results 1 - 10 of about 491,000 for "kernel methods".

- Still work to do
  - Results 1 - 10 of about 65,300,000 for "Britney Spears".

Outline

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

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.
PCA

\[
D = \sum_{i=1}^{d} (d_i - \mu) \approx BC + \mu 1^T
\]

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

\[
d^T D^T B = B \Lambda
\]

**Error Function for PCA**

- PCA minimizes the following **CONVEX** function.

\[
E(B,C) = \sum_{i=1}^{d} \| d_i - Bc_i \|^2 = \| D - BC \|^2.
\]

- Not unique solution: \( B^T C = BC \) \( R \in \mathbb{R}^{k \times k} \)

- To obtain same PCA solution \( R \) has to satisfy:

\[
\hat{B} = BR \quad \hat{C} = R^{-1}C
\]

\[
\hat{B}^T \hat{B} = I \quad \hat{C} \hat{C}^T = \Lambda
\]

- \( R \) is computed as a generalized k x k eigenvalue problem. (de la Torre, 2006)

\[
(C \Lambda)^{-1} R = B^T B \Lambda A^{-1}
\]

**Snap-shot Method & SVD**

- If \( d >> n \) (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 \( D \)).

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

\[
d^T D^T B = B \Lambda
\]

**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, 1998)
  - Appearance/shape models (Cootes & Taylor, 2001; Cootes et al., 1998; Pentland, 1986; Shum et al., 1995; McKenna et al., 1997; de la Torre et al., 1998b; de la Torre, 2006)
  - **Structure from Motion** (Tomasi & Kanade, 1992; Bregler et al., 2000; Sturm & Triggs, 1996; Brand, 2001)
  - **Bilinear models** (Tenenbaum & Freeman, 2000; Marimont & Wandell, 1992)
  - **Direct extensions** (Welling et al., 2003; Penev & Aick, 1996)
  - **Image watermarking** (Lu & Tan, 2000)
  - **Signal processing** (Moonen & de Moor, 1995)
  - **Neural approaches** (Cija, 1999; Sanger, 1988; Xu, 1993)
  - **Bilinear models** (Tenenbaum & Freeman, 2000; Marimont & Wandell, 1992)

**SVD factorizes the data matrix \( D \) as:**

\[
D = U \Sigma V^T
\]

\[
D^T = V \Lambda U^T
\]
“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

\[
\min_{B \geq 0, C \geq 0} F = \sum_y \left| d_y - (BC)_y \right|^2
\]

Inference:

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

Learning:

\[
B_v = \frac{(DC^T)_v}{(B C^T)_v}
\]

- Multiplicative algorithm can be interpreted as diagonally rescaled gradient descent.
Independent Component Analysis

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

ICA

(Hyvärinen et al., 2001)

\[ \mathbf{D} = \mathbf{B} \mathbf{C} \]

\[ \mathbf{C} \approx \mathbf{S} = \mathbf{W} \mathbf{D} \]

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

ICA vs PCA

Many optimization criteria

- Minimize high order moments: e.g. kurtosis
  \[ \text{kurt}(\mathbf{W}) = \mathbb{E}(s^4) - 3(\mathbb{E}(s^2))^2 \]
- Many other information criteria.
- Also an error function: (Olshausen & Field, 1996)
  \[ \sum_{i=1}^{n} |d_i - \mathbf{B}c_i| + \sum_{i=1}^{n} S(c_i) \]  \text{Sparseness (e.g. S=|\cdot|)}
- Other sparse PCA.  
  (Chennubhotla & Jepson, 2001b; Zou et al., 2005; dAspremont et al., 2004.)
Basis of natural images

Denoising

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

Multidimensional Scaling (MDS)
- MDS takes a matrix of pair-wise distances and finds an embedding that preserves the interpoint distances.

Optimize w.r.t. y_i
MDS(II)

Criterion is invariant w.r.t rotations and translations. However, it is not invariant to scaling. Better criterion is
\[ \sum \sum (d_i - d_j)^2 \]
\[ \sum d_i^2 \]
Called stress

An example: map of the US

Observed distance between points i and j in p-space
Distance between the points in two-dimensional space
Mapping
Component Analysis for Signal Processing

Outline

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

Linear Discriminant Analysis (LDA)

(Fisher, 1936; Mardia et al., 1979; Bishop, 1995)

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

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

Oriented Component Analysis (OCA)

- Generalized eigenvalue problem: $\Sigma_{i} b_k = \Sigma_{e} b_k \lambda$
- $b_{oca}$ is steered by the distribution of noise.
OCA for face recognition

Canonical Correlation Analysis (CCA)

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

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 \( X \in \mathbb{R}^{d_x \times n} \) and \( Y \in \mathbb{R}^{d_y \times n} \), CCA finds the pair of directions \( w_x \) and \( w_y \) that maximize the correlation between the projections (assume zero mean data)

\[
\rho = \frac{w'_x X’ Y w'_y}{\sqrt{w'_x X’ X w'_x} \sqrt{w'_y Y’ Y w'_y}}
\]

- Several ways of optimizing it:

\[
A = \begin{bmatrix} 0 & X'Y \\ X'Y & 0 \end{bmatrix} \in \mathbb{R}^{(d_x+d_y)^2 \times (d_x+d_y)^2}, \quad B = \begin{bmatrix} X'X & 0 \\ 0 & Y'Y \end{bmatrix} \in \mathbb{R}^{(d_x+d_y)^2 \times (d_x+d_y)^2} \quad w = \begin{bmatrix} w_x' \\ w_y' \end{bmatrix}
\]

- An stationary point of \( r \) is the solution to CCA.
\[
Aw = \lambda Bw
\]
Robot localization with Canonical Correlation Analysis
(Skocaj & Leonardis, 2000)

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

Kernel Methods

Linear methods fail
- When data points sit on a non-linear manifold
  - We won't find a good linear mapping from the data points to a plane, because there isn't any
  - In the end, linear methods do nothing more than rotate/translate/scale data
Linear methods fail

- Learning a non-linear representation for classification

Kernel Methods

- Suppose $\phi(.)$ is given as follows
  $$\phi\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) = (1, \sqrt{2}x_1, \sqrt{2}x_2, x_1^2, x_2^2, \sqrt{2}x_1x_2)$$

- An inner product in the feature space is
  $$\langle \phi\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right), \phi\left(\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}\right)\rangle = (1 + x_1y_1 + x_2y_2)^2$$

- So, if we define the kernel function as follows, there is no need to carry out $\phi(.)$ explicitly
  $$K(x, y) = (1 + x_1y_1 + x_2y_2)^2$$

- This use of kernel function to avoid carrying out $\phi(.)$ explicitly is known as the kernel trick. In any linear algorithm that can be expressed by inner products can be made nonlinear by going to the feature space

Kernel Methods for Classification

- The kernel defines an implicit mapping (usually high dimensional and non-linear) from input to feature space, so the data becomes linearly separable.
- Computation in the feature space can be costly because it is (usually) high dimensional
  - The feature space is typically infinite-dimensional!

Kernel PCA

(Scholkopf et al., 1998)
Kernel PCA
(Schölkopf et al., 1998)

- Eigenvectors of the cov. Matrix in feature space.
\[
\mathbf{C} = \frac{1}{n} \sum_{i=1}^{n} \Phi(d_i)\Phi(d_i)^T \quad \mathbf{C} \mathbf{b}_i = \mathbf{b}_i \lambda_i
\]
- Eigenvectors lie in the span of data in feature space.
\[
\mathbf{b}_i = \sum_{i=1}^{n} \alpha_i \Phi(d_i)
\]
\[
\frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \Phi(d_i) K(d_i, d_j) = \left[ \sum_{i=1}^{n} \alpha_i \Phi(d_i) \right] \lambda_i
\]
\[
K\mathbf{a} = \mathbf{a} \lambda_i
\]

Factor Analysis

- A Gaussian distribution on the coefficients and noise is added to PCA\rightarrow Factor Analysis, (Mardia et al., 1979)
\[
d = \mu + Bc + \eta
\]
\[
p(c, d) = N(c | 0, I) \quad p(d | c, B) = N(d | \mu + Bc, \Psi)
\]
\[
p(\eta) = N(\eta | 0, \Psi) \quad \Psi = \text{diag}(\eta_1, \eta_2, \ldots, \eta_k)
\]
\[
\text{cov}(d) = \text{E}((d - \mu)(d - \mu)^T) = B B^T + \Psi
\]
- Inference (Roweis & Ghahramani, 1999; Tipping & Bishop, 1999a)
\[
p(c, d) \quad \text{Jointly Gaussian}
\]
\[
p(c | d) = N(c | m, V)
\]
\[
m = B^T (B B^T + \Psi)^{-1} (d - \mu)
\]
\[
V = (I + B^T \Psi^{-1} B)^{-1}
\]

PPPc

- If \( \Psi = \text{E}(\eta \eta^T) = d_1 \) PPCA.
- If \( \varepsilon \rightarrow 0 \) is equivalent to PCA\rightarrow 0 \quad B'(BB' + \Psi)^{-1} = (B' B)^{-1} B'
- Probabilistic visual learning (Moghaddam & Pentland, 1997)
\[
p(d) = \int p(d | c) p(c | d) \text{d}c = \frac{1}{(2\pi)^{d/2} |\Psi|^{1/2}} \exp\left\{-\frac{1}{2} (d - \mu)^T (\Psi + B B^T)^{-1} (d - \mu)\right\}
\]
\[
c_i = B^T d_i
\]
Tensor Factorization

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

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

Data Organization
- Linear/PCA: Data Matrix
  - $R$-pixels x images
  - a matrix of image vectors
- Multilinear: Data Tensor $D$
  - $R$-people x views x illums x express x pixels
  - N-dimensional matrix
  - 28 people, 45 images/person
  - 5 views, 3 illuminations, 3 expressions per person
N-Mode SVD Algorithm

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

\[ N = 3 \]

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

<table>
<thead>
<tr>
<th></th>
<th>TensorFaces</th>
<th>PCA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Sq. Err.</td>
<td>409.15</td>
<td>85.75</td>
</tr>
<tr>
<td>3 illum + 11 people param.</td>
<td>33 basis vectors</td>
<td>33 basis vectors</td>
</tr>
</tbody>
</table>

Acknowledgments

- The content of some of the slides has been taken from previous presentations/papers of:
  - Ales Leonardis.
  - Horst Bischof.
  - Michael Black.
  - Rene Vidal.
  - Anat Levin.
  - Juha Karhunen.
  - Andrew Fitzgibbon.
  - Daniel Lee.
  - Chris Ding.
  - M. Alex Vasilescu.
  - Sam Roweis.
  - Daoqiang Zhang.
  - Ammon Shashua.
Bibliography

Bibliography

(No specific content provided in the image)
A bibliography from a technical paper on component analysis for signal processing.