A Unifying View of Component Analysis Methods (from a Computer Vision Perspective)

Fernando De la Torre

Outline

- Introduction
- Generative models
  - Principal Component Analysis (PCA).
  - Non-negative Matrix Factorization (NMF).
  - Independent Component Analysis (ICA).
- Discriminative models
  - Linear Discriminant Analysis (LDA).
  - Canonical Correlation Analysis (CCA).
- Standard extensions of linear models
  - Latent variable models.
  - Kernel methods.
  - Tensor factorization
- Other extensions
  - Filtered Component Analysis (active appearance models)
  - Learning Kernel expansions (support vector machines)
  - Parameterized cluster analysis (clustering)

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, visualization, dimensionality reduction
- Signal Processing
  - Spectral estimation, system identification (e.g. Kalman filter), cocktail problem, eco cancellation
- Computer Graphics
  - Compression (BRDF), synthesis,…
- Speech, bioinformatics, combinatorial problems.
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, visualization, dimensionality reduction

- Signal Processing
  - Spectral analysis, time-frequency analysis (e.g. Kalman filter).
  - Sensor array processing (e.g. cocktail problem, echo cancelation, ...)

- Computer Graphics
  - Compression (BRDF), synthesis, ... 
  - Speech, bioinformatics, combinatorial problems.

Why Component Analysis?

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

- Easy to incorporate
  - Robustness to noise, missing data, outliers (Torre & Black 03)
  - Invariance to geometric transformations (Torre et al. 07, Torre & Black 02)
  - Non-linear (kernel methods) (Vapnik 95, Scholkopf’99, Cristian & Taylor’04)
  - Probabilistic (latent variable models) (Everitt 84, Bishop et al 99, Benter’80)
  - Multi-factorial (tensors) (Ralu’88, Vasilescu & Terzopoulos ‘02, Avidan & Shashua’97)
  - Exponential family PCA (Gordon 02)

- Efficient methods $O(n \cdot d \cdot \log n)$

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".

Generative Models

$D \approx BC$
**Principal Component Analysis (CA)**

- **SVD** (Beltrami, 1873; Jordan 1874); **PCA** (Pearson, 1901; Hotelling, 1933)

\[
DD^T B = BB\Lambda
\]

**Part-based representation**

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

**Optimization for PCA**

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

\[
E(B, C) = \sum_{i=1}^{n} \|d_i - Bc_i\|_2^2 = \|D - BC\|_F
\]
Non-negative Matrix Factorization

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

Independent Component Analysis

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

ICA vs PCA

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

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)
  \[ \sum_{i=1}^{n} \| d_i - Bc_i \| + \sum_{i=1}^{n} S(c_i) \]
  Sparseness (e.g. \( S = 1 \))
- Other sparse PCA.
  (Chennubhotla & Jepson, 2001b; Zou et al., 2005; dAspremont et al., 2004)
Discriminative Models

- Linear Discriminant Analysis (LDA)
- Canonical Correlation Analysis (CCA)

Linear Discriminant Analysis (LDA)

(Fisher, 1938; Mardia et al., 1979; Bishop, 1995)

\[ S_b = \sum_{i=1}^{C} \sum_{j=1}^{K} (\mu_i - \mu_J)(\mu_i - \mu_J)^T \]

\[ S_i = DD^T = \sum_{i=1}^{C} d_i d_i^T \]

\[ \lambda(B) = \begin{bmatrix} S_s B & S_s B \end{bmatrix} \]

\[ S_b = S_s B A \]

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

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

\[ \rho = \frac{w_x^T X^T w_y}{\sqrt{w_x^T X^T w_x w_y^T Y^T w_y}} \]

- Several ways of optimizing it:

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

\[ w = \begin{bmatrix} w_x \\ w_y \end{bmatrix} \]

- An stationary point of \( r \) is the solution to CCA.

\[ Aw = \lambda Bw \]

- PCA independently and general mapping

- Signals dependent signals with small energy can be lost.
Standard extensions

- Tensor Factorization
- Latent Variable Models
- Kernel Methods

Tensor Factorization

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

- People
- Expressions
- Views
- Illuminations

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**
  - \( R \text{ pixels} \times \text{images} \)
  - a matrix of image vectors

- **Multilinear: 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

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

\( N = 3 \)

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
Latent Variable Models

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

Ppca
- If \( \Psi = E(\eta \eta^T) = d_A \), PPCA.
- If \( \varepsilon \to 0 \) is equivalent to PCA. \( \varepsilon \to 0 \) \( B^T (BB^T + \Psi)^{-1} = (B^T B)^{-1}B^T \)
- Probabilistic visual learning (Moghaddam & Pentland, 1997)
  \[
  p(d) = \int p(d \mid c) p(c) dc = \int e^{-\frac{1}{2} (d - \mu)^T \Sigma^{-1} (d - \mu)} e^{-\frac{1}{2} (c - \mu)^T \Sigma^{-1} (c - \mu)}
  \]
  \[
  \Sigma = \left[ \begin{array}{cc}
  \Sigma_{cc} & \Sigma_{cd} \\
  \Sigma_{dc} & \Sigma_{dd}
  \end{array} \right]
  \]
  \[
  c_j = B^T d_j
  \]

Other non-linear
- Mixture of Factor Analyzers
- Generative topographic mapping
Kernel methods

- Non-linear high dimensional mapping of the data so it becomes linearly separable in the feature space.
- This is equivalent to finding a non-linear decision boundary in the input space.
- Computation in the feature space can be costly because it is high dimensional.
  - The feature space is typically infinite-dimensional!

Examples

\[ \langle \mathbf{b}, \Phi(\mathbf{d}) \rangle = \sum_{i=1}^{n} \alpha_i K(\mathbf{d}_i, \mathbf{d}) \]

Clinical Depression

- Point Prevalence
  - 10% of the population
- High personal and social costs
- Current standards for diagnosis or therapeutic effectiveness of treatment rely on:
  - Interview with the physician.
  - Self-report measures.

Second Part
Behavioral Analysis from Audio and Video

Automatic FACS coding

Problems to Solve

Classification
Au classification

Lips are relaxed and parted.

Kernels

- Support Vector Machines
  - Large margin (QP- no local minima).
- Kernels.

\[(x_1, x_2) \rightarrow (x_1, x_2, x_1^2, x_2^2) = (z_1, z_2, z_3)\]

Input space  Feature space

Problem

- Learning a non-linear representation for classification

Previous work

- Neural networks
  - Hard to train, local minima, difficult interpretation of hidden units.
- Metric learning (King et al. '02, Goldberger et al. '05, He et al. '03, Weinberger '06, Lebanon '03, Shental '02)
- Kernels (Cristiani et al. '01, Lanchkriet et al. '04, Chapelle et al. '02, Kwok et al. '03)
  - Learning kernel expansions
  - Kernel with dimensionality reduction
  - Computational cost \(O(dnk^2)\) samples features
Learning Kernel Expansions

\[ K(B_1, \ldots, B_p, \lambda_1, \ldots, \lambda_p, \alpha) = \sum_{i=0}^{k} \alpha_i \hat{M}_i^T \alpha_i > 0 \]

\[ k_{ij} = \sum_{i=0}^{k} \alpha_i (\hat{m}_{ij}) \alpha_i > 0 \]

\[ \hat{m}_{ij} = \left[ \frac{d_i^T (B B^T \lambda_i \alpha_i) d_i}{\sqrt{d_i^T (B B^T \alpha_i) d_i d_i^T (B B^T \lambda_i \alpha_i) d_i}} \right] \]

Learning from an Ideal Kernel

\[ E = \| W \odot (F - K(B_1, \ldots, B_p, \lambda_1, \ldots, \lambda_p, \alpha)) \|_F \]

- Avoids over-fitting.
- Outliers/missing data
- Unbalanced classes.

First order (t=1) \[ E = \| F - D^{T} B B^T D \|_F \] LDA

CA a unifying view

Experiments

- Training (20,000) Cross validation (10,000) Testing (20,000)

Pedestrian Detection

- Linear SVM
- RBF SVM

CA a unifying view
Clustering

The Clustering Problem

- Partition the data set in c-disjoint “clusters” of data points.

- Number of possible partitions

\[ S(n,c) = \frac{1}{c!} \sum_{i=0}^{c} (-1)^i \binom{c}{i} n^i \quad \left( \begin{array}{c} n = 21 \\ c = 4 \end{array} \right) \approx 10^{12} \]

- NP-hard and approximate algorithms (k-means, hierarchical clustering, mog, …)

Clustering AUs

- How to cluster invariantly to geometric transformations to capture subtle facial behavior?

K-means

- K-means criterion:

\[
E(B, G^T) = \left\| d_1 \ldots d_n - [b_1 \ldots b_n] G^T \right\|_F
\]

- Problems:
  - Local minima and sensitive to initial conditions
  - Optimal for spherical clusters
A Unifying View of Clustering

- **K-means** (Gu et al. '01) \( E(G, B) = \| D - BG^T \|_F \)
- Energy-based Spectral Clustering (Chilson et al., '04, Torre et al. '06).
  \[ E(G, B) = \sum \text{Normalized Cuts} (Shi & Malik '00) \]
  \[ \Gamma = \{ \varphi(d_1), \varphi(d_2), \ldots, \varphi(d_n) \} \]
  \[ \Gamma = \varphi(D) \]

Parameterized Cluster Analysis

\[ E_1(G, B) = \| \Gamma - BG^T \|_F \]
\[ \Gamma = \{ \varphi(d_1), \varphi(d_2), \ldots, \varphi(d_n) \} \]

- Parameterize the shape.
  \[ \Gamma = \{ \varphi(A, d_1), \varphi(A, d_2), \ldots, \varphi(A, d_n) \} \]

\[ E_2(A, G) = \text{tr}(\Gamma(A)^T (I - G^T G^{-1} G^T)) \]
\[ G \geq 0 \quad G1_e = 1_n \]

Example
An Energy-based Framework for Component Analysis

**Modeling**
- Filtered CA [CVPR-Torre et al. 07]
- Robust PCA [UCV-Torre & Black 03]
- Convolutional CA [Torre et al. In prep.]
- Parameterized CA [VVC-Torre & Black 03]
- Unsupervised bilinear [ICCV-Gross et al. 07]

**Classification**
- Learning kernel [CVPR-Torre & Vinyals 07]
- Multimodal Oriented DA [ICML-Torre et al. 05]
- Robust PCA [IJCV-Torre & Black 03]
- Parameterized CA [IVC-Torre & Black 03]

**Clustering**
- Parameterized CA [CVPR-Torre et al. 07]

**Visualization**
- DCCA [CVPR-Torre & Black 02]
- Dynamic MRS [ICML-Cabero et al. 07]

- $\mathcal{L}(B, C) = \|D - BC\|_F$
- $\mathcal{L}(B, A) = \sum_{i} D B B D_{[\text{normalization}]} - I$

- $\mathcal{L}(D(\tilde{f}(\cdot, \text{a})) \sim \text{BG}) W_f$
Thanks

1st Workshop on Component Analysis

http://www.cs.cmu.edu/~ftorre/ca/