Recognition III

Introduction to Computer Vision

CSE 152

Lecture 20

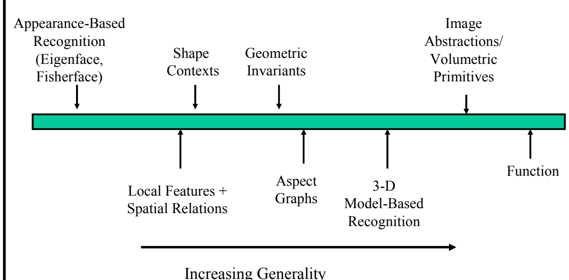
Announcements

- Assignment 5: Due Friday, 4:00
- Final Exam: Wed, 6/9/04, 11:30-2:30, WLH 2207 (here)
- I'll discuss briefly today, and will be at discussion section tomorrow for first 45 minutes.

Virtual Cinematography: *Making 'The Matrix' Sequels*
George Borshukov
VFX Technology Supervisor, ESC Entertainment
Friday, June 4, 2004
1:00 p.m. to 2:30 p.m.
[Pizza lunch will precede the event from noon to 1 p.m.]
Main Auditorium, San Diego Supercomputer Center

The presentation will cover the key technologies that had to be developed and deployed to create the synthetic human sequences in the Matrix sequels including Universal Capture - image-based facial animation, realistic human face rendering, and use of measured BRDF in film production. It will also feature a breakdown of The Superpunch shot (pictured above) from "*The Matrix Revolutions*" (the bullet time punch that Neo delivers to Agent Smith during the film's last face-off). This difficult, important, expensive, and challenging shot was entirely computer generated and showcased the technological developments of 3.5+ years at their best by showing a full-frame close-up of a known human actor.

A Rough Recognition Spectrum

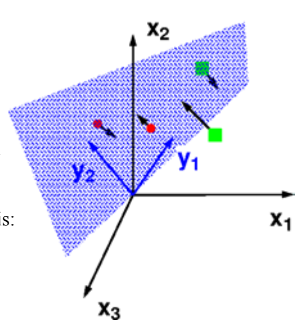


Projection, and reconstruction

- An n-pixel image $x \in \mathbf{R}^n$ can be projected to a low-dimensional feature space $y \in \mathbf{R}^m$ by

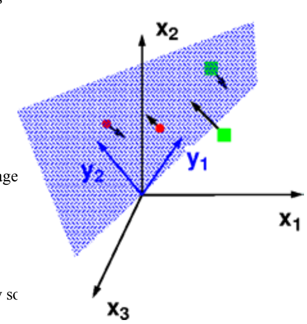
$$y = Wx$$

- From $y \in \mathbf{R}^m$, the reconstruction of the point is $\mathbf{W}^T\mathbf{y}$
- The error of the reconstruction is: $||\mathbf{x}-\mathbf{W}^T\mathbf{W}\mathbf{x}||$



Face detection using "distance to face space"

- Scan a window ω across the image, and classify the window as face/not face as follows:
- Project window to subspace, and reconstruct as described earlier.
- Compute distance between ω and reconstruction.
- Local minima of distance over all image locations less than some treshold are taken as locations of faces.
- Repeat at different scales.
- Possibly normalize windows intensity sc that |ω| = 1.



Singular Value Decomposition

- Any m by n matrix A may be factored such that

$$A = U\Sigma V^T$$

$$[m \times n] = [m \times m][m \times n][n \times n]$$

- U: m by m, orthogonal matrix
  - Columns of **U** are the eigenvectors of AA^T
- V: n by n, orthogonal matrix,
  - columns are the eigenvectors of A^TA
- **Σ:** m by n, diagonal with non-negative entries ($\sigma_1, \sigma_2, \ldots, \sigma_s$) with s=min(m,n) are called the called the singular values
  - **Singular values are the square roots of eigenvalues of both** AA^T **and** A^TA ***& Columns of U are corresponding Eigenvectors!!***
  - *Result of SVD algorithm:* $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_s$

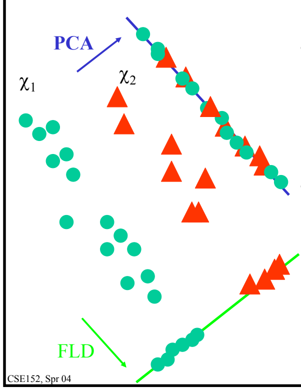
Performing PCA with SVD

- Singular values of A are the square roots of eigenvalues of both AA^T and A^TA & Columns of U are corresponding Eigenvectors
- And $\sum_{i=1}^{n} a_i a_i^T = [a_1 \quad a_2 \quad \cdots \quad a_n][a_1 \quad a_2 \quad \cdots \quad a_n]^T = AA^T$
- Covariance matrix is:

$$\Sigma = \frac{1}{n}\sum_{i=1}^{n}(\vec{x}_i - \vec{\mu})(\vec{x}_i - \vec{\mu})^T$$

- So, ignoring 1/n subtract mean image μ from each input image, create data matrix, and perform (thin) SVD on the data matrix.

PCA & Fisher's Linear Discriminant



- PCA (Eigenfaces)

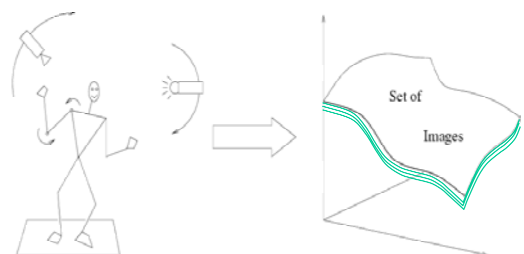
$$\boxed{W_{PCA} = \arg\max_W \left|W^T S_T W\right|}$$

Maximizes projected total scatter

- Fisher's Linear Discriminant

$$\boxed{W_{fld} = \arg\max_W \frac{\left|W^T S_B W\right|}{\left|W^T S_W W\right|}}$$

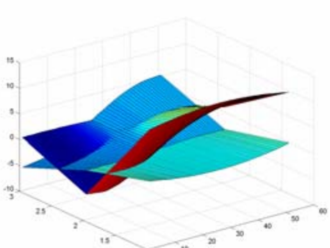
Maximizes ratio of projected between-class to projected within-class scatter



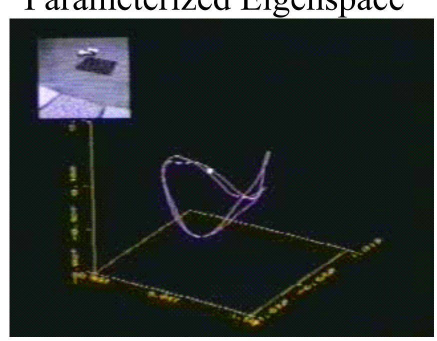
Variability: Camera position
Illumination
Internal parameters

Within-class variations

An example: surfaces of first 3 coefficients



Parameterized Eigenspace



Bayesian Classification

Discussed on blackboard, but slides may be helpful

Basic ideas in classifiers

- Loss
  - some errors may be more expensive than others
    - e.g. a fatal disease that is easily cured by a cheap medicine with no side-effects -> false positives in diagnosis are better than false negatives
  - We discuss two class classification: L(1->2) is the loss caused by calling 1 a 2
- Total risk of using classifier s

$$R(s) = Pr\left\{1 \to 2|\text{using } s\right\} L(1 \to 2) + Pr\left\{2 \to 1|\text{using } s\right\} L(2 \to 1)$$

Basic ideas in classifiers

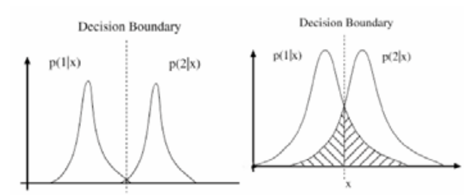
- Generally, we should classify as 1 if the expected loss of classifying as 1 is better than for 2
- gives

1 if $\quad p(1|\boldsymbol{x})L(1 \to 2) > p(2|\boldsymbol{x})L(2 \to 1)$

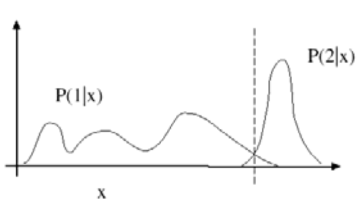
2 if $\quad p(1|\boldsymbol{x})L(1 \to 2) < p(2|\boldsymbol{x})L(2 \to 1)$

- Crucial notion: Decision boundary
  - points where the loss is the same for either case

Some loss may be inevitable: the minimum risk (shaded area) is called the Bayes risk



Finding a decision boundary is not the same as modelling a conditional density.



Example: known distributions

$$p(\underline{x}|k) = \left(\frac{1}{2\pi}\right)^{-p/2} \left|\underline{\underline{\Sigma}}\right|^{-1/2} \exp\left[-\frac{1}{2}\left(\underline{x}-\underline{\mu}_k\right)^T \underline{\underline{\Sigma}}^{-1}\left(\underline{x}-\underline{\mu}_k\right)\right]$$

- Assume normal class densities, p-dimensional measurements with common (known) covariance and different (known) means
- Class priors are π_k
- Can ignore a common factor in posteriors - important; posteriors are then:

$$p(k|x) \propto (\pi_k)\left(\frac{1}{2\pi}\right)^{-p/2} \left|\underline{\Sigma}\right|^{-1/2} \exp\left[-\frac{1}{2}\left(x-\underline{\mu}_k\right)^T \underline{\Sigma}^{-1}\left(x-\underline{\mu}_k\right)\right]$$

- Classifier boils down to: choose class that minimizes:

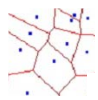
$$\delta(\underline{x}, \underline{\mu}_k)^2 - 2 \log \pi_k$$

where

Mahalanobis distance

$$\delta(\underline{x}, \underline{\mu}_k) = \left[(\underline{x} - \underline{\mu}_k)^T \underline{\Sigma}^{-1} (\underline{x} - \underline{\mu}_k) \right]^{1/2}$$

because covariance is common, this simplifies to sign of a linear expression (i.e. Voronoi diagram in 2D for Σ=I)



Finding skin

- Skin has a very small range of (intensity independent) colours, and little texture
  - Compute an intensity-independent colour measure, check if colour is in this range, check if there is little texture (median filter)
  - See this as a classifier - we can set up the tests by hand, or learn them.
  - get class conditional densities (histograms), priors from data (counting)
- Classifie
  - if $p(\text{skin}|\boldsymbol{x}) > \theta$, classify as skin
  - if $p(\text{skin}|\boldsymbol{x}) < \theta$, classify as not skin
  - if $p(\text{skin}|\boldsymbol{x}) = \theta$, choose classes uniformly and at random

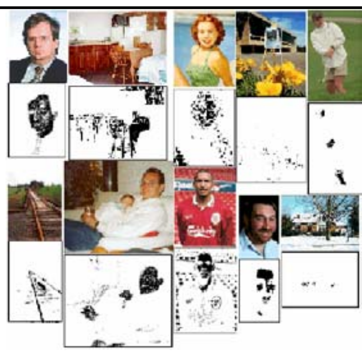


Figure from "Statistical color models with application to skin detection," M.J. Jones and J. Rehg, Proc. Computer Vision and Pattern Recognition, 1999 copyright 1999, IEEE

Receiver **O**perating **C**urve

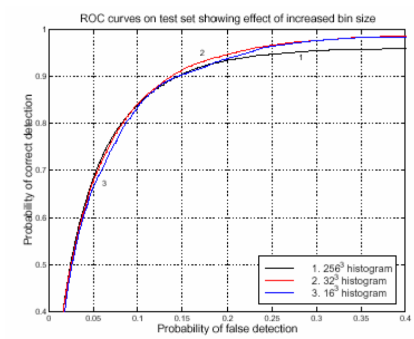


Figure from "Statistical color models with application to skin detection," M.J. Jones and J. Rehg, Proc. Computer Vision and Pattern Recognition, 1999 copyright 1999, IEEE

Appearance-Based Vision: Lessons

Strengths

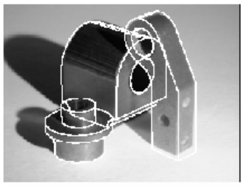
- Posing the recognition metric in the image space rather than a derived representation is more powerful than expected.
- Modeling objects from many images is not unreasonable given hardware developments.
- The data (images) may provide a better representations than abstractions for many tasks.

Appearance-Based Vision: Lessons

Weaknesses

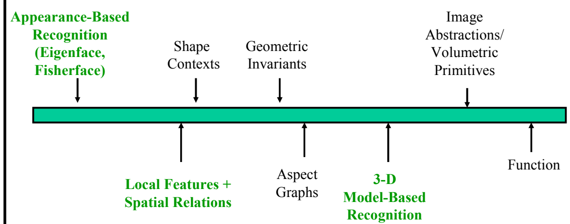
- Segmentation or object detection is still an issue.
- To train the method, objects have to be observed under a wide range of conditions (e.g. pose, lighting, shape deformation).
- Limited power to extrapolate or generalize (abstract) to novel conditions.

Model-Based Vision



- Given 3-D models of each object
- Detect image features (often edges, line segments, conic sections)
- Establish correspondence between model &image features
- Estimate pose
- Consistency of projected model with image.

A Rough Recognition Spectrum



Recognition by Hypothesize and Test

- General idea
  - Hypothesize object identity and pose
  - Recover camera parameters (widely known as backprojection)
  - Render object using camera parameters
  - Compare to image
- Issues
  - where do the hypotheses come from?
  - How do we compare to image (verification)?
- Simplest approach
  - Construct a correspondence for all object features to every correctly sized subset of image points
    - These are the hypotheses
  - Expensive search, which is also redundant.

Pose consistency

- Correspondences between image features and model features are not independent.
- A small number of correspondences yields a camera matrix --- the others correspondences must be consistent with this.
- Strategy:
  - Generate hypotheses using small numbers of correspondences (e.g. triples of points for a calibrated perspective camera, etc., etc.)
  - Backproject and verify

Scene Interpretation

"The Swing"
Fragonard, 1766

Final Exam

- Closed book
- One cheat sheet
  - Single piece of paper, handwritten, no photocopying, no physical cut & paste. – you can start with sheet from the midterm, if you want.
- What to study
  - Basically material presented in class, and supporting material from text
  - If it was in text, but NEVER mentioned in class, it is very unlikely to be on the exam
- Question style:
  - Short answer
  - Some longer problems to be worked out.