

Recognition Wrap Up and onto Motion

Introduction to Computer Vision
CSE 152
Lecture 19a

CSE152, Winter 2013

Intro Computer Vision

Announcements

- HW4 (face recognition) has been posted.
- Final exam: Tuesday 3/19 8:00AM
- Fill out your CAPES

CSE152, Winter 2013

Intro Computer Vision

Recognition: What have we seen

- Within-class vs. Between class variability
- Sliding window classifiers
- Image as a feature vector, but other features can be used too.
- Three methods for using linear subspaces for classification
 - Eigenfaces – principal component analysis (PCA)
 - Linear subspace modeling variation of each class (lighting)
 - Fisherfaces
- Singular Value Decomposition (SVD)

CSE152, Winter 2013

Intro Computer Vision

Linear Subspaces & Linear Projection

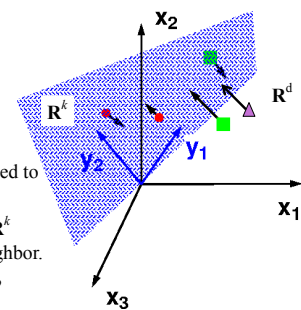
- A d -pixel image $x \in \mathbf{R}^d$ can be projected to a low-dimensional feature space $y \in \mathbf{R}^k$ by

$$y = Wx$$

where W is an k by d matrix.

- Each training image is projected to the subspace
- Recognition is performed in $\mathbf{R}^k$ using, for example, nearest neighbor.
- How do we choose a good W ?

Example: Projecting from $\mathbf{R}^3$ to $\mathbf{R}^2$



CSE152, Winter 2013

Intro Computer Vision

Eigenfaces

Modeling

1. Given a collection of n training images x_i , represent each one as a d -dimensional column vector
2. Compute the mean image and covariance matrix.
3. Compute k Eigenvectors of the covariance matrix corresponding to the k largest Eigenvalues and form matrix $W^T = [v_1, v_2, \dots, v_k]$ (Or perform using SVD!!)
 - Note that the Eigenvectors are images
4. Project the training images to the k -dimensional Eigenspace. $y_i = Wx_i$

Recognition

1. Given a test image x , project the vectorized image to the Eigenspace by $y = Wx$
2. Perform classification of y to the projected training images.

CSE152, Winter 2013

Intro Computer Vision

Performing PCA with SVD

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

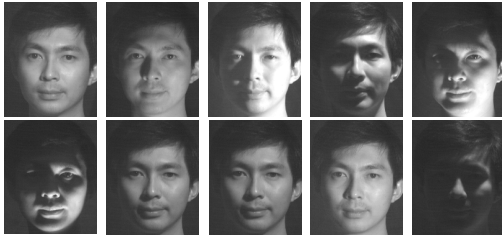
$$\Sigma = \frac{1}{n} \sum_{i=1}^n (\bar{x}_i - \bar{\mu})(\bar{x}_i - \bar{\mu})^T$$

- So, ignoring $1/n$ subtract mean image μ from each input image, create a $d \times n$ data matrix, and perform thin SVD on the data matrix. $D = [x_1 - \mu \mid x_2 - \mu \mid \dots \mid x_n - \mu]$

CSE152, Winter 2013

Intro Computer Vision

Illumination Variability



- How does the set of images of an object in fixed pose vary as lighting changes?
- How can we recognize people across all lighting conditions without having to see the person every way?

CSE152, Winter 2013

Intro Computer Vision

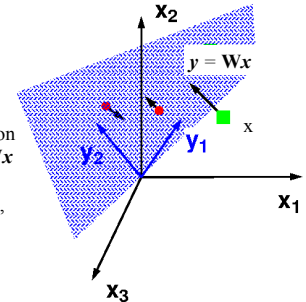
Distance to Linear Subspace

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

$$y = Wx$$

- From $y \in \mathbb{R}^k$, the reconstruction of the point in $\mathbb{R}^d$ is $W^T y = W^T W x$

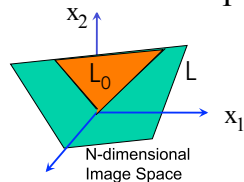
- The error of the reconstruction, or the distance from x to the subspace spanned by W is: $\|x - W^T W x\|$



CSE152, Winter 2013

Intro Computer Vision

Distance to Linear Subspace



- From a collection of images of person i under variable lighting, compute basis B_i images of a 3-D linear subspace using PCA or SVD.
- Recognize individual by finding subspace with shortest distance from a test image to the subspace.

$$d_i = \|x - B_i B_i^T x\|$$

CSE152, Winter 2013

Intro Computer Vision

PCA & Fisher's Linear Discriminant

- Between-class scatter

$$S_B = \sum_{i=1}^c |\chi_i| (\mu_i - \mu)(\mu_i - \mu)^T$$

- Within-class scatter

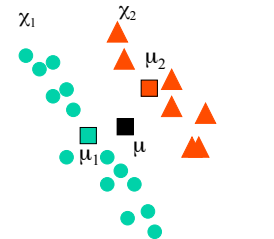
$$S_W = \sum_{i=1}^c \sum_{x_i \in \chi_i} (x_i - \mu_i)(x_i - \mu_i)^T$$

- Total scatter

$$S_T = \sum_{i=1}^c \sum_{x_i \in \chi_i} (x_i - \mu)(x_i - \mu)^T = S_B + S_W$$

- Where

- c is the number of classes
- μ_i is the mean of class χ_i
- $|\chi_i|$ is number of samples of χ_i .

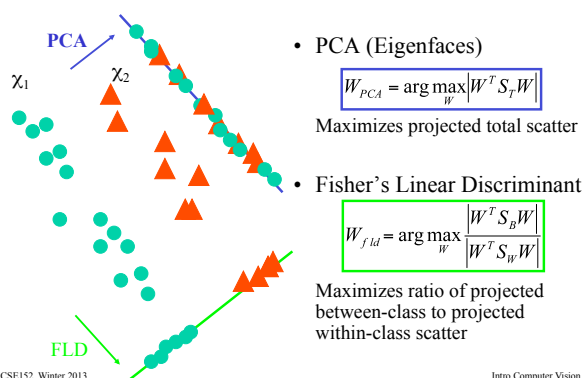


If the data points x_i are projected by $y_i = Wx_i$ and the scatter of x_i is S , then the scatter of the projected points y_i is $W^T S W$

CSE152, Winter 2013

Intro Computer Vision

PCA & Fisher's Linear Discriminant



- PCA (Eigenfaces)

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

Maximizes projected total scatter

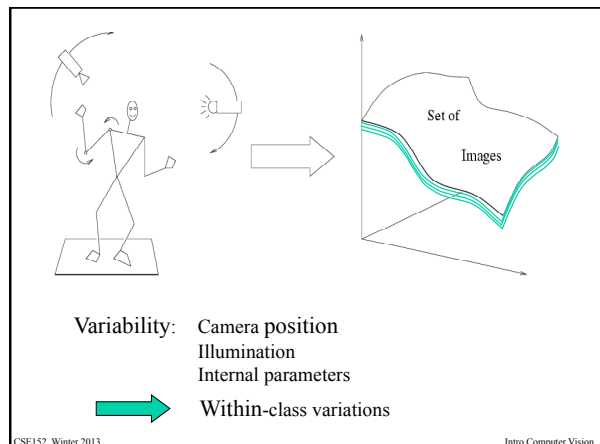
- Fisher's Linear Discriminant

$$W_{FLD} = \arg \max_W \frac{|W^T S_B W|}{|W^T S_W W|}$$

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

CSE152, Winter 2013

Intro Computer Vision



Variability: Camera position
Illumination
Internal parameters

Within-class variations

CSE152, Winter 2013

Intro Computer Vision

Appearance manifold approach

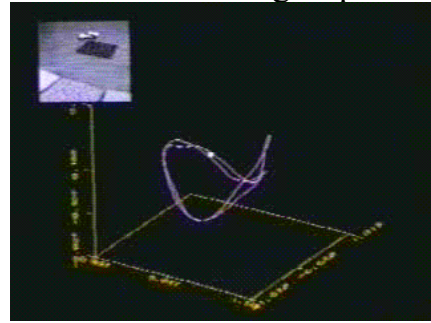
- for every object (Nayar et al. '96)
 1. sample the set of viewing conditions
 2. Crop & scale images to standard size
 3. Use as feature vector
- apply a PCA over all the images
- keep the dominant PCs
- Set of views for one object is represented as a manifold in the projected space
- Recognition: What is nearest manifold for a given test image?



CSE152, Winter 2013

Intro Computer Vision

Parameterized Eigenspace



CSE152, Winter 2013

Intro Computer Vision

Recognition

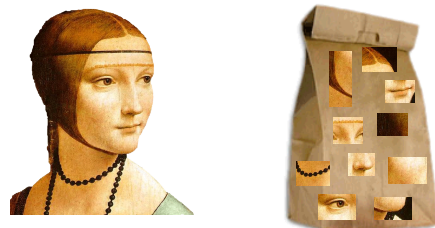


CSE152, Winter 2013

Intro Computer Vision

Bag-of-features models

Object → Bag of 'words'



CSE152, Winter 2013

Slides from Svetlana Lazebnik who borrowed from others

Intro Computer Vision

Bag-of-features models

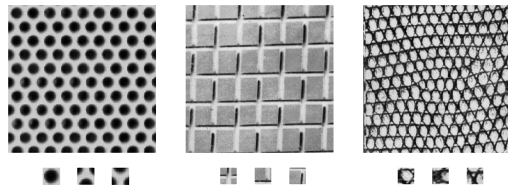


CSE152, Winter 2013

Intro Computer Vision

Origin 1: Texture recognition

- Texture is characterized by the repetition of basic elements or *textons*
- For stochastic textures, it is the identity of the textons, not their spatial arrangement, that matters



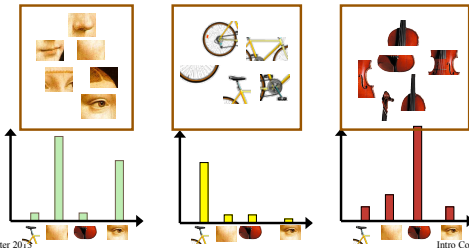
Julesz, 1981; Cula & Dana, 2001; Leung & Malik 2001; Mori, Belongie & Malik, 2001; Schmid 2001; Varma & Zisserman, 2002, 2003; Lazebnik, Schmid & Ponce, 2003

CSE152, Winter 2013

Intro Computer Vision

Bag-of-features steps

1. Extract features
2. Learn “visual vocabulary”
3. Quantize features using visual vocabulary
4. Represent images by frequencies (histogram) of “visual words”
5. Recognition using histograms as input to classifier



CSE152, Winter 2009

Intro Computer Vision

Feature extraction

- Regular grid or interest regions



CSE152, Winter 2013

Intro Computer Vision

Even without shading, shape reveals a lot.
Would Bag of words capture this?



CSE152, Winter 2013

Intro Computer Vision

Motion

Introduction to Computer Vision
CSE 152
Lecture 19b

CSE152, Winter 2013

Intro Computer Vision

Motion

What are problems that we solve using motion

1. Correspondence: Where have elements of the image moved between image frames
2. Ego Motion: How has the camera moved.
3. Reconstruction: Given correspondence, what is 3-D geometry of scene
4. Segmentation: What are regions of image corresponding to different moving objects
5. Tracking: Where have objects moved in the image? related to correspondence and segmentation.

Variations:

- Small motion (video),
- Wide-baseline (multi-view)

CSE152, Winter 2013

Intro Computer Vision

Structure-from-Motion (SFM)

Goal: Take as input two or more images or video w/o any information on camera position/motion, and estimate camera position and 3-D structure of scene.

Two Approaches

1. Discrete motion (wide baseline)
 1. Orthographic (affine) vs. Perspective
 2. Two view vs. Multi-view
 3. Calibrated vs. Uncalibrated
2. Continuous (Infinitesimal) motion

CSE152, Winter 2013

Intro Computer Vision

Discrete Motion: Some Counting

Consider M images of N points, how many unknowns?

1. Camera locations: Affix coordinate system to location of first camera location: $(M-1)*6$ Unknowns
2. 3-D Structure: $3*N$ Unknowns
3. Can only recover structure and motion up to scale. Why?

Total number of unknowns: $(M-1)*6+3*N-1$

Total number of measurements: $2*M*N$

Solution is possible when $(M-1)*6+3*N-1 \leq 2*M*N$

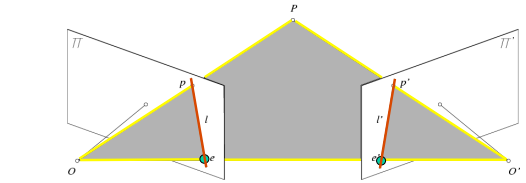
$M=2 \rightarrow N \geq 5$

$M=3 \rightarrow N \geq 4$

CSE152, Winter 2013

Intro Computer Vision

Epipolar Constraint: Calibrated Case



$$\vec{O_p} \cdot [\vec{OO'} \times \vec{O'p'}] = 0 \Rightarrow \mathbf{p} \cdot [\mathbf{t} \times (\mathbf{R}\mathbf{p}')] = 0 \quad \text{with} \quad \begin{cases} \mathbf{p} = (u, v, 1)^T \\ \mathbf{p}' = (u', v', 1)^T \\ \mathbf{M} = (\mathbf{Id} \quad \mathbf{0}) \\ \mathbf{M}' = (\mathbf{R}^T, -\mathbf{R}^T\mathbf{t}) \end{cases}$$

Essential Matrix
(Longuet-Higgins, 1981)

$$\mathbf{p}^T \mathbf{E} \mathbf{p}' = 0 \quad \text{with} \quad \mathbf{E} = [\mathbf{t}_\times] \mathbf{R}$$

$$\text{where } [\mathbf{t}_\times] = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}$$

CSE152, Winter 2013

Intro Computer Vision

The Eight-Point Algorithm (Longuet-Higgins, 1981)

Let $\mathbf{F}$ denote the Essential Matrix here

$$(u, v, 1) \begin{pmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{pmatrix} \begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix} = 0 \Rightarrow (uu', uv', u, vu', vv', v, u', v', 1) \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix} = 0$$

Set F_{33} to 1

$$\begin{pmatrix} u_1 u'_1 & u_1 v'_1 & u_1 & v_1 u'_1 & v_1 v'_1 & v_1 & u'_1 & v'_1 \\ u_2 u'_2 & u_2 v'_2 & u_2 & v_2 u'_2 & v_2 v'_2 & v_2 & u'_2 & v'_2 \\ u_3 u'_3 & u_3 v'_3 & u_3 & v_3 u'_3 & v_3 v'_3 & v_3 & u'_3 & v'_3 \\ u_4 u'_4 & u_4 v'_4 & u_4 & v_4 u'_4 & v_4 v'_4 & v_4 & u'_4 & v'_4 \\ u_5 u'_5 & u_5 v'_5 & u_5 & v_5 u'_5 & v_5 v'_5 & v_5 & u'_5 & v'_5 \\ u_6 u'_6 & u_6 v'_6 & u_6 & v_6 u'_6 & v_6 v'_6 & v_6 & u'_6 & v'_6 \\ u_7 u'_7 & u_7 v'_7 & u_7 & v_7 u'_7 & v_7 v'_7 & v_7 & u'_7 & v'_7 \\ u_8 u'_8 & u_8 v'_8 & u_8 & v_8 u'_8 & v_8 v'_8 & v_8 & u'_8 & v'_8 \end{pmatrix} \begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

Solve For $\mathbf{F}$

Solve For $\mathbf{R}$ and $\mathbf{t}$

CSE152, Winter 2013

Intro Computer Vision

Sketch of Two View SFM Algorithm

Input: Two images

1. Detect feature points
2. Find 8 matching feature points (easier said than done, but usually done with RANSAC Algorithm)
3. Compute the Essential Matrix $\mathbf{E}$ using Normalized 8-point Algorithm
4. Compute $\mathbf{R}$ and $\mathbf{T}$ (recall that $\mathbf{E}=\mathbf{R}\mathbf{S}$ where $\mathbf{S}$ is skew symmetric matrix)
5. Perform stereo matching using recovered epipolar geometry expressed via $\mathbf{E}$.
6. Reconstruct 3-D geometry of corresponding points.

CSE152, Winter 2013

Intro Computer Vision

Continuous Motion

- Consider a video camera moving continuously along a trajectory (rotating & translating).
- How do points in the image move
- What does that tell us about the 3-D motion & scene structure?



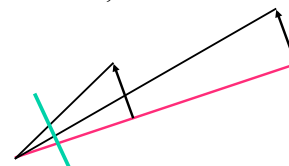
CSE152, Winter 2013

Intro Computer Vision

Motion

“When objects move at equal speed, those more remote seem to move more slowly.”

- Euclid, 300 BC



CSE152, Winter 2013

Intro Computer Vision

Simplest Idea for video processing

Image Differences: Brightness change at a location

- Given image $I(u,v,t)$ and $I(u,v, t+\delta t)$, compute $I(u,v, t+\delta t) - I(u,v,t)$.
- This is partial derivative: $\frac{\partial I}{\partial t}$
- At object boundaries, $\left| \frac{\partial I}{\partial t} \right|$ is large and is a cue for segmentation
- Doesn't tell which way stuff is moving

CSE152, Winter 2013

Intro Computer Vision

Background Subtraction

- Gather image $I(x,y,t_0)$ of background without objects of interest (perhaps computed over average over many images).
- At time t , pixels where $|I(x,y,t) - I(x,y,t_0)| > \tau$ are labeled as coming from foreground objects



Raw Image



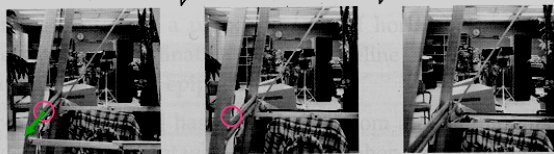
Foreground region

CSE152, Winter 2013

Intro Computer Vision

The Motion Field

Where in the image did a point move?

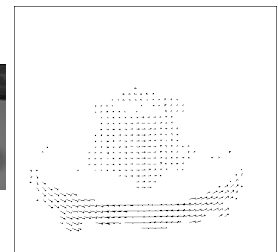
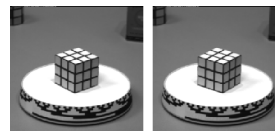


Down and left

CSE152, Winter 2013

Intro Computer Vision

Motion Field



CSE152, Winter 2013

Intro Computer Vision

What causes a motion field?

1. Camera moves (translates, rotates)
2. Object's in scene move rigidly
3. Objects articulate (pliers, humans, quadripeds)
4. Objects bend and deform (fish)
5. Blowing smoke, clouds

CSE152, Winter 2013

Intro Computer Vision