

Recognition I

Introduction to Computer Vision
CSE 152
Lecture 17

CSE152, Winter 2013

Intro Computer Vision

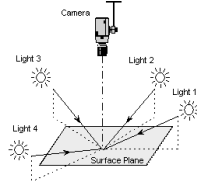
Announcements

- HW 3 due on Thursday.
- HW4 to be posted shortly

CSE152, Winter 2013

Intro Computer Vision

Two shape-from-X methods that use shading

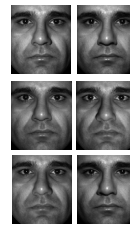


- Photometric stereo: Single viewpoint, multiple images under different lighting.
 1. Arbitrary known BRDF
 2. Lambertian BRDF, known lighting
 3. Lambertian BRDF, unknown lighting.

CSE152, Winter 2013

Intro Computer Vision

An example of photometric stereo



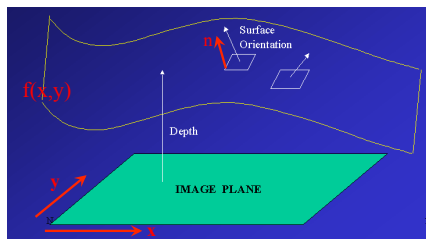
Input Images



CSE152, Winter 2013

Intro Computer Vision

Gradient Space (p,q)



Gradient Space : (p,q)

$$p = \frac{\partial f}{\partial x}, \quad q = \frac{\partial f}{\partial y}$$

Normal vector

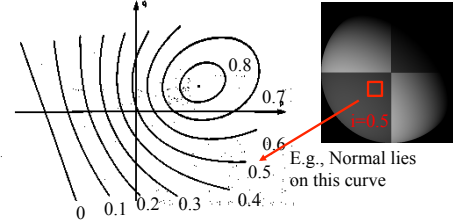
$$\mathbf{n} = \frac{\partial s}{\partial x} \mathbf{x} + \frac{\partial s}{\partial y} \mathbf{y} = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, -1 \right)^T$$

$$\hat{\mathbf{n}} = \frac{1}{\sqrt{p^2 + q^2 + 1}} (p, q, -1)^T$$

CSE152, Winter 2013

Intro Computer Vision

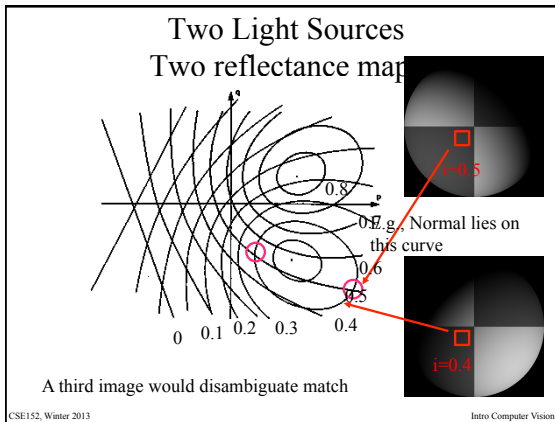
Reflectance Map of Lambertian Surface



What does the intensity (Irradiance) of one pixel in one im
It constrains the surface normal projecting to that point to

CSE152, Winter 2013

Intro Computer Vision



Recovering the surface $f(x,y)$

Many methods: Simplest approach

1. From normal field $\mathbf{n} = (n_x, n_y, n_z)$, $p = -n_x/n_z$, $q = -n_y/n_z$
2. Integrate $p = df/dx$ along a row $(x, 0)$ to get $f(x, 0)$
3. Then integrate $q = df/dy$ along each column starting with value of the first row

CSE152, Winter 2013

Intro Computer Vision

What might go wrong?

- Height $z(x,y)$ is obtained by integration along a curve from (x_0, y_0) .

$$z(x,y) = z(x_0, y_0) + \int_{(x_0, y_0)}^{(x,y)} (p dx + q dy)$$

- If one integrates the derivative field along any closed curve, one expects to get back to the starting value.
- Might not happen because of noisy estimates of (p,q)

CSE152, Winter 2013

Intro Computer Vision

What might go wrong?

Integrability. If $z(x,y)$ is the height function, we expect that

$$\frac{\partial}{\partial y} \frac{\partial z}{\partial x} = \frac{\partial}{\partial x} \frac{\partial z}{\partial y}$$

In terms of estimated gradient space (p,q) , this means:

$$\frac{\partial p}{\partial y} = \frac{\partial q}{\partial x}$$

But since p and q were estimated independently at each point as intersection of curves on three reflectance maps, equality is not going to exactly hold

CSE152, Winter 2013

Intro Computer Vision

II. Photometric Stereo: Lambertian Surface, Known Lighting

CSE152, Winter 2013

Intro Computer Vision

Lambertian Surface

At image location (u,v) , the intensity of a pixel $x(u,v)$ is:

$$e(u,v) = [a(u,v) \hat{\mathbf{n}}(u,v)] \cdot [s_0 \hat{\mathbf{s}}] = \mathbf{b}(u,v) \cdot \mathbf{s}$$

where

- $a(u,v)$ is the albedo of the surface projecting to (u,v) .
- $\mathbf{n}(u,v)$ is the direction of the surface normal.
- s_0 is the light source intensity.
- $\mathbf{s}$ is the direction to the light source.

CSE152, Winter 2013

Intro Computer Vision

Lambertian Photometric stereo

- If the light sources s_1 , s_2 , and s_3 are **known**, then we **can** recover b from as few as three images. (Photometric Stereo: Silver 80, Woodham81).

$$[e_1 \ e_2 \ e_3] = b^T [s_1 \ s_2 \ s_3]$$

- i.e., we measure e_1 , e_2 , and e_3 and we know s_1 , s_2 , and s_3 . We can then solve for b by solving a linear system.

$$b^T = [e_1 \ e_2 \ e_3] [s_1 \ s_2 \ s_3]^{-1}$$

- Normal is: $n = b/|b|$, albedo is: $|b|$

- Let E_1, E_2, E_3 be vectorized images, we can solve for all normals, scaled by the albedos as:

$$B = [E_1 \ E_2 \ E_3] [s_1 \ s_2 \ s_3]^{-1}$$

CSE152, Winter 2013

Intro Computer Vision

III. Photometric Stereo with unknown lighting and Lambertian surfaces

CSE152, Winter 2013

Intro Computer Vision

How do you construct subspace?

$$[E_1 \ E_2 \ E_3] = B^T [s_1 \ s_2 \ s_3]$$

$[e_1 \ e_2 \ e_3]$ B

- Given three or more images $E_1 \dots E_n$, estimate B and s_i .
- How? Given images in form of $E = [E_1 \ E_2 \ \dots]$, Compute SVD(E) and let B^* be the n by 3 matrix formed by first 3 singular values.

CSE152, Winter 2013

Intro Computer Vision

Matrix Decompositions

- Definition: The factorization of a matrix M into two or more matrices $M_1, M_2, \dots, M_n$, such that $M = M_1 M_2 \dots M_n$.
- Many decompositions exist...
 - QR Decomposition
 - LU Decomposition
 - LDU Decomposition
 - Etc.

CSE152, Winter 2013

Intro Computer Vision

Singular Value Decomposition

Excellent ref: 'Matrix Computations,' Golub, Van Loan

- 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 $A^T A$
- Σ : m by n , diagonal with non-negative entries ($\sigma_1, \sigma_2, \dots, \sigma_s$) with $s = \min(m, n)$ are called the singular values
 - Singular values are the square roots of eigenvalues of both AA^T and $A^T A$
 - Result of SVD algorithm: $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_s$

CSE152, Winter 2013

Intro Computer Vision

Applying SVD to Photometric stereo

- The images are formed by

$$[e_1 \ e_2 \ e_3 \ \dots \ e_n] = B^T [s_1 \ s_2 \ s_3 \ \dots \ s_n]$$

$$E = B^T S$$
- So, $\text{svd}(E) = U \Sigma V^T$ where U is N by n , Σ is n by n , and V^T is n by N
- Without noise, we expect 3 non-zero singular values, and so $U \Sigma V^T = U' \Sigma' V'^T$ where U' is N by 3, Σ' is 3 by 3, and V'^T is 3 by n .
- In particular $B = U' A$ where A is some 3×3 matrix.

CSE152, Winter 2013

Intro Computer Vision

Do Ambiguities Exist? Yes

- Is B unique?
- For any invertible matrix A , $B^* = BA$ also a solution
- For any image of B produced with light source S , the same image can be produced by lighting $B^* = BA$ with $S^* = A^{-1}S$ because

$$X = B^*S^* = BAA^{-1}S = BS$$
- When we estimate B using SVD, the rows are NOT generally the normal times the albedo.

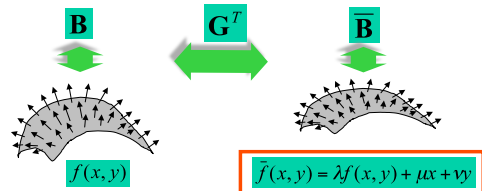
CSE152, Winter 2013

Intro Computer Vision

GBR Transformation

Only **Generalized Bas-Relief** transformations satisfy the integrability constraint:

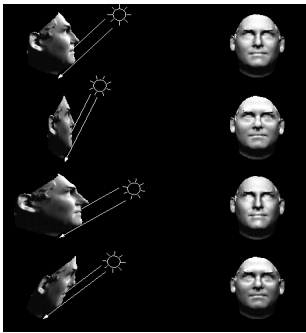
$$A = G^T = \begin{bmatrix} \lambda & 0 & -\mu \\ 0 & \lambda & -\nu \\ 0 & 0 & 1 \end{bmatrix}$$



CSE152, Winter 2013

Intro Computer Vision

Generalized Bas-Relief Transformations



Without knowledge of light source location, one can only recover surfaces up to GBR transformations.

$$\bar{f}(x, y) = \lambda f(x, y) + \mu x + \nu y$$

f : true depth

$\bar{f}$: GBR transform of depth

CSE152, Winter 2013

Intro Computer Vision

Uncalibrated photometric stereo

- Take n images as input without knowledge of light directions or strengths
- Perform SVD to compute B^* .
- Find some A such that B^*A is close to integrable.
- Integrate resulting gradient field to obtain height function $f^*(x, y)$.

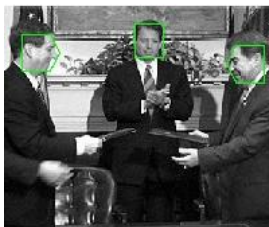
Comments:

- $f^*(x, y)$ differs from $f(x, y)$ by a GBR.
- Can use specularities to resolve GBR for non-Lambertian surface.

CSE152, Winter 2013

Intro Computer Vision

Recognition

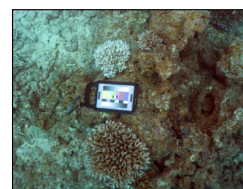


Given a database of objects and an image determine what, if any of the objects are present in the image.

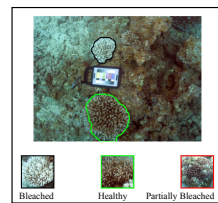
CSE152, Winter 2013

Intro Computer Vision

Where are the coral heads and which ones are healthy and which are bleached?



Input Image

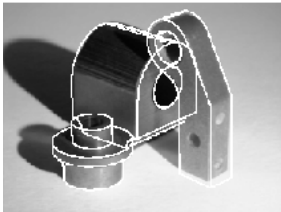


Segmented/labelled Image

CSE152, Winter 2013

Intro Computer Vision

Recognition



Given a database of objects and an image determine what, if any of the objects are present in the image.

CSE152, Winter 2013

Intro Computer Vision

Recognition

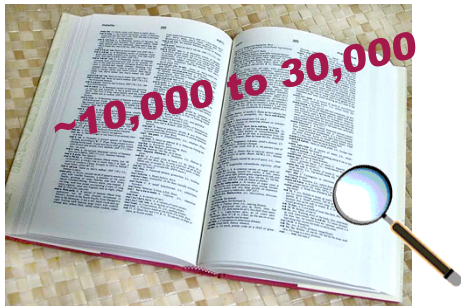


Given a database of objects and an image determine what, if any of the objects are present in the image.

CSE152, Winter 2013

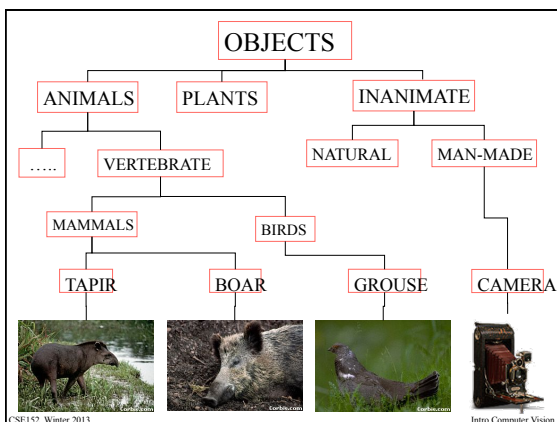
Intro Computer Vision

How many visual object categories are there?



CSE152, Winter 2013

Biederman, 198



CSE152, Winter 2013

Intro Computer Vision

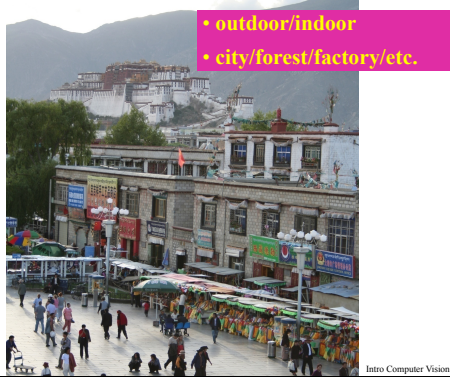
Specific recognition tasks



CSE152, Winter 2013

Intro Computer Vision

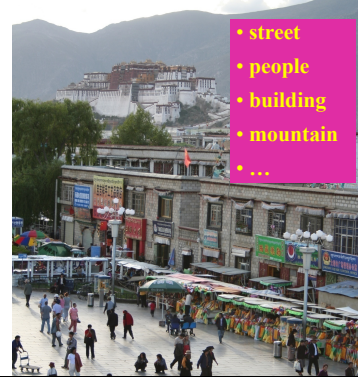
Scene categorization



CSE152, Winter 2013

Intro Computer Vision

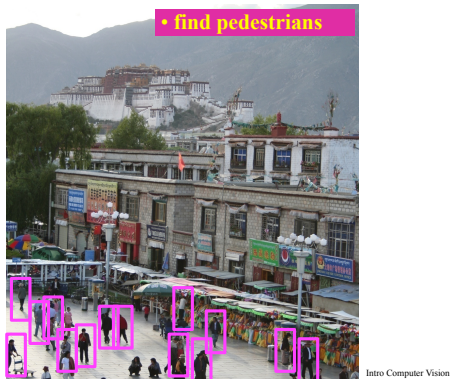
Image annotation/tagging



CSE152, Winter 2013

Intro Computer Vision

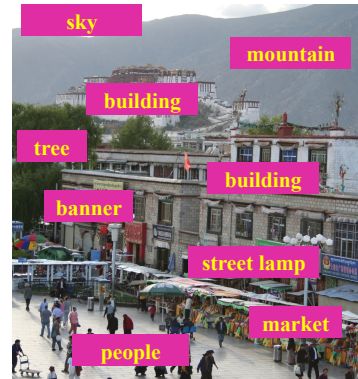
Object detection



CSE152, Winter 2013

Intro Computer Vision

Image parsing



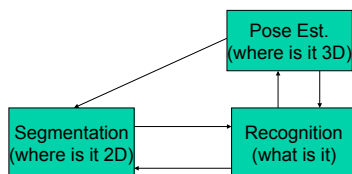
CSE152, Winter 2013

Intro Computer Vision

Object Recognition: The Problem

Given: A database D of "known" objects and an image I:

1. Determine which (if any) objects in D appear in I
2. Determine the pose (rotation and translation) of the object



WHAT AND WHERE!!!

CSE152, Winter 2013

Intro Computer Vision

Within-class variations



CSE152, Winter 2013

Intro Computer Vision

Recognition Challenges

- Within-class variability
 - Different objects within the class have different shapes or different material characteristics
 - Deformable
 - Articulated
 - Compositional
- Pose variability:
 - 2-D Image transformation (translation, rotation, scale)
 - 3-D Pose Variability (perspective, orthographic projection)
- Lighting
 - Direction (multiple sources & type)
 - Color
 - Shadows
- Occlusion – partial
- Clutter in background -> false positives

CSE152, Winter 2013

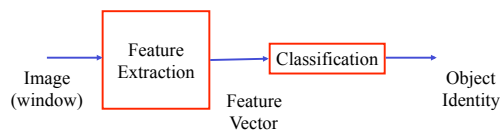
Intro Computer Vision

- Categories near top of tree (e.g., vehicles) – lots of within class variability
- Fine grain categories (e.g., species of birds) -- Moderate within class variation
- Instance recognition (e.g., person identification) – within class mostly shape articulation, bending, etc.

CSE152, Winter 2013

Intro Computer Vision

Sketch of a Pattern Recognition Architecture



CSE152, Winter 2013

Intro Computer Vision

Sliding window approaches



CSE152, Winter 2013

Intro Computer Vision

Example: Face Detection

- Scan window over image.
- Search over position & scale.
- Classify window as either:
 - Face
 - Non-face



CSE152, Winter 2013

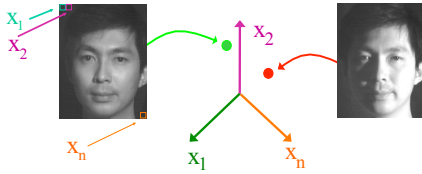
Intro Computer Vision

- So, what are the features?
- So, what is the classifier

CSE152, Winter 2013

Intro Computer Vision

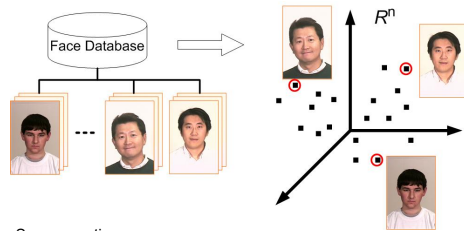
The Space of Images



- We will treat an d -pixel image as a point in an d -dimensional space, $\mathbf{x} \in \mathbb{R}^d$.
- Each pixel value is a coordinate of $\mathbf{x}$.

Face Space:

- A set of face images construct a *face space* in $\mathbb{R}^d$
- Appearance-based methods analyze the distributions of individual faces in face space



Some questions:

1. How are images of an individual, under all conditions, distributed in this space?
2. How are the images of all individuals distributed in this space?

CSE152

Computer Vision I

More features

- Filtered image
- Filter with multiple filters (bank of filters)
- Histogram of colors
- Histogram of Gradients (HOG)
- Haar wavelets
- Scale Invariant Feature Transform (SIFT)
- Speeded Up Robust Feature (SURF)

CSE152

Computer Vision I

- So, what are the features?
- So, what is the classifier?

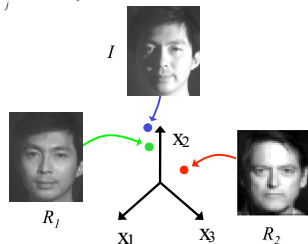
CSE152

Computer Vision I

Nearest Neighbor Classifier

$\{R_j\}$ are set of training images.

$$ID = \underset{j}{\operatorname{argmin}} \operatorname{dist}(R_j, I)$$



CSE152, Winter 2013

Intro Computer Vision

Comments on Nearest Neighbor

- Sometimes called "Template Matching"
- Variations on distance function (e.g. L_1 , robust distances)
- Multiple templates per class- perhaps many training images per class.
- Expensive to compute k distances, especially when each image is big (d -dimensional).
- May not generalize well to unseen examples of class.
- No worse than twice the error rate of the optimal classifier -- if enough training samples
- Some solutions:
 - Bayesian classification
 - Dimensionality reduction

CSE152, Winter 2013

Intro Computer Vision

Do features vectors have structure in the image space?

- Faces of individuals cluster in the image space. (Not true).
- Faces of individuals are confined to a linear or affine subspace of $\mathbf{R}^d$
- Faces of an individual are approximated by a linear subspace
- Faces and objects lie on or near a manifold in the space of images.

CSE152

Computer Vision I

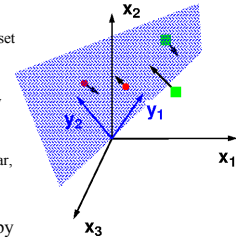
An idea:

Represent the set of images as a linear subspace

What is a linear subspace?

Let V be a vector space and let W be a subset of V . Then W is a subspace iff:

1. The zero vector, 0 , is in W .
2. If u and v are elements of W , then any linear combination of u and v is an element of W ; $au + bv \in W$
3. If u is an element of W and c is a scalar, then the scalar product $cu \in W$



A k -dimensional subspace is spanned by k linearly independent vectors

It is spanned by a k -dimensional orthogonal basis.

Example: A 2-D linear subspace of $\mathbf{R}^3$ spanned by y_1 and y_2 .

CSE152

Computer Vision I

Linear Subspaces & Linear Projection

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

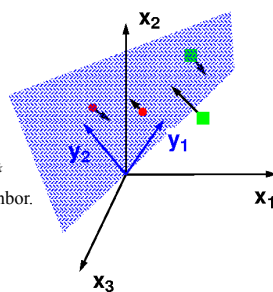
$$y = Wx$$

where W is an k by d matrix.

• 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

Linear Subspaces & Recognition

1. Approximate all training images as a single linear subspace (Eigenfaces).
2. Represent lighting variation w/o shadowing for a single individual as a 3-D linear subspace. A collection of n individuals is modeled as n 3-D linear subspaces.
3. Represent lighting variation w/ shadowing for a single individual as a 9-D linear subspace. A collection of n individuals is modeled as n 9-D linear subspaces.
4. Project all training images to a single subspace that enhances discriminability (Fisherfaces).

CSE152, Winter 2013

Intro Computer Vision

Eigenfaces: Principal Component Analysis (PCA)

Assume we have a set of n feature vectors x_i ($i = 1, \dots, n$) in $\mathbf{R}^d$. Write

$$\mu = \frac{1}{n} \sum_i x_i$$

$$\Sigma = \frac{1}{n-1} \sum_i (x_i - \mu)(x_i - \mu)^T$$

The unit eigenvectors of Σ — which we write as $v_1, v_2, \dots, v_d$, where the order is given by the size of the eigenvalue and v_1 has the largest eigenvalue — give a set of features with the following properties:

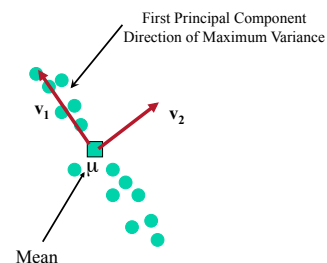
- They are independent.
- Projection onto the basis $\{v_1, \dots, v_k\}$ gives the k -dimensional set of linear features that preserves the most variance.

Algorithm 22.5: Principal components analysis identifies a collection of linear features that are independent, and capture as much variance as possible from a dataset.

CSE152, Winter 2013

Intro Computer Vision

PCA Example



CSE152, Winter 2013

Intro Computer Vision

Eigenfaces

- **Modeling**
 1. Given a collection of n training images x_i , represent each one as d -dimensional column vector
 2. Compute the mean image and covariance matrix.
 3. Compute k Eigenvectors of covariance matrix corresponding to k largest Eigenvalues and form matrix $W^T = [v_1, v_2, \dots, v_k]$ (Or perform using SVD!!) (note that these are images)
 4. Project the training images to the k -dimensional Eigenspace. $y_i = W^T x_i$
- **Recognition**
 1. Given a test image x , project vectorized image to Eigenspace by $y = W^T x$
 2. Perform classification to the projected training images.