Recap: Image Segmentation

- Goal: identify groups of pixels that go together

Recap: Images as Graphs

- Fully-connected graph
  - Node (vertex) for every pixel
  - Link between every pair of pixels, \((p,q)\)
  - Affinity weight \(w_{pq}\) for each link (edge)

Recap: Normalized Cut (NCut)

- A minimum cut penalizes large segments
- This can be fixed by normalizing for size of segments
- The normalized cut cost is:
  \[ Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)} \]
  \[ assoc(A,V) = \text{sum of weights of all edges in } V \text{ that touch } A \]
  \[ = \text{cut}(A,B) \left[ \sum_{p \in A} w_{p,q} + \sum_{q \in B} w_{p,q} \right] \]
- The exact solution is NP-hard but an approximation can be computed by solving a generalized eigenvalue problem.

Recap: NCuts: Overall Procedure

1. Construct a weighted graph \(G=(V,E)\) from an image.
2. Connect each pair of pixels, and assign graph edge weights,
   \[ W(i,j) = \text{Prob. that } i \text{ and } j \text{ belong to the same region} \]
3. Solve \((D-W)\mathbf{y} = \lambda D\mathbf{y}\) for the smallest few eigenvectors. This yields a continuous solution.
4. Threshold eigenvectors to get a discrete cut
   - This is where the approximation is made (we’re not solving NP).
5. Recursively subdivide if NCut value is below a pre-specified value.
Recap: Energy Formulation

- Energy function
  \[ E(x, y) = \sum_{i,j} \phi(x_i, y_j) + \sum_{i,j} \psi(x_i, x_j) \]
  - Single-node potentials \( \phi \)
    - Encode local information about the given pixel/patch
    - How likely is a pixel/patch to belong to a certain class (e.g. foreground/background)?
  - Pairwise potentials \( \psi \)
    - Encode neighborhood information
    - How different is a pixel/patch’s label from that of its neighbor? (e.g. based on intensity/color/texture difference, edges)

Recap: Graph Cuts Energy Minimization

- Regional bias example
  Suppose \( I \) and \( I' \) are given “expected” intensities of object and background
  - \( D_s(i) = \exp(-\|I_i - I'_i\|^2 / 2\sigma^2) \)
  - \( D_s(i) = \exp(-\|I_i - I'_i\|^2 / 2\sigma^2) \)

Recap: When Can s-t Graph Cuts Be Applied?

- \( s-t \) graph cuts can only globally minimize binary energies that are submodular.
  \[ E(L) \text{ can be minimized by } s-t \text{ graph cuts} \]
  \[ E(L) \leq E(s,t) + E(t,s) \]
  - Non-submodular cases can still be addressed with some optimality guarantees.
  - Current research topic
Topics of This Lecture
- Subspace Methods for Recognition
  - Motivation
- Principal Component Analysis (PCA)
  - Derivation
  - Object recognition with PCA
  - Eigenimages/Eigenfaces
  - Limitations
- Fisher’s Linear Discriminant Analysis (LDA)
  - Derivation
  - Fisherfaces for recognition
- Robust PCA
  - Application: robot localization

Recap: Appearance-Based Recognition
- Basic assumption
  - Objects can be represented by a set of images (“appearances”).
  - For recognition, it is sufficient to just compare the 2D appearances.
  - No 3D model is needed.

Recap: Recognition Using Global Features
- E.g. histogram comparison

Representations for Recognition
- More generally, we want to obtain representations that are well-suited for
  - Recognizing a certain class of objects
  - Identifying individuals from that class (identification)
- How can we arrive at such a representation?
- Approach 1:
  - Come up with a brilliant idea and tweak it until it works.
  - Can we do this more systematically?

Example: The Space of All Face Images
- When viewed as vectors of pixel values, face images are extremely high-dimensional
  - 100x100 image = 10,000 dimensions
- However, relatively few 10,000-dimensional vectors correspond to valid face images
- We want to effectively model the subspace of face images

The Space of All Face Images
- We want to construct a low-dimensional linear subspace that best explains the variation in the set of face images
Subspace Methods

- Images represented as points in a high-dim. vector space
- Valid images populate only a small fraction of the space
- Characterize subspace spanned by images

Image set → Basis images → Representation coefficients

Slide credit: Ales Leonardis

Topics of This Lecture

- Subspace Methods for Recognition
  - Motivation
- Principal Component Analysis (PCA)
  - Derivation
  - Object recognition with PCA
  - Eigenvectors/Eigenfaces
  - Limitations
- Fisher’s Linear Discriminant Analysis (LDA)
  - Derivation
  - Fisherfaces for recognition
- Robust PCA
  - Applications: robot localization

Principal Component Analysis

- Given: N data points \( x_1, \ldots, x_N \) in \( \mathbb{R}^d \)
- We want to find a new set of features that are linear combinations of original ones:
  \[ u(x_i) = u^T (x_i - \mu) \]
  (\( \mu \): mean of data points)
- What unit vector \( u \) in \( \mathbb{R}^d \) captures the most variance of the data?

Remember: Fitting a Gaussian

- Mean and covariance matrix of data define a Gaussian model

\[
\begin{align*}
\text{var}(u) &= \frac{1}{N} \sum_{i=1}^{N} u^T (x_i - \mu)(u^T (x_i - \mu))^T \\
&= \frac{1}{N} u^T \left[ \sum_{i=1}^{N} (x_i - \mu)(x_i - \mu)^T \right] u \\
&= \frac{1}{N} u^T \Sigma u
\end{align*}
\]
- The direction that maximizes the variance is the eigenvector associated with the largest eigenvalue of \( \Sigma \).
Interpretation of PCA

- Compute eigenvectors of covariance, $S$
- Eigenvectors: main directions
- Eigenvalue: variance along eigenvector

Result: coordinate transform to best represent the variance of the data.

Properties of PCA

- It can be shown that the mean square error between $x_i$ and its reconstruction using only $m$ principle eigenvectors is given by the expression:
  $$\sum_{j=1}^{m} \lambda_j - \sum_{j=1}^{m} \lambda_j = \sum_{j=m+1}^k \lambda_j$$

Interpretation
  - PCA minimizes reconstruction error
  - PCA maximizes variance of projection
  - Finds a more “natural” coordinate system for the sample data.

Projection and Reconstruction

- An $n$-pixel image $x \in \mathbb{R}^n$ can be projected to a low-dimensional feature space $y \in \mathbb{R}^m$ by $y = Wx$
- From $y \in \mathbb{R}^m$, the reconstruction of the point is $Wy$
- The error of the reconstruction is $\|x - W^TWx\|

Example: Object Representation

Object Detection by Distance TO Eigenspace

- Scan a window $\omega$ over the image and classify the window as object or non-object as follows:
  - Project window to subspace and reconstruct as earlier.
  - Compute the distance between $\omega$ and the reconstruction (reprojection error).
  - Local minima of distance over all image locations $\Rightarrow$ object locations
  - Repeat at different scales
  - Possibly normalize window intensity such that $|\omega|=1.$
Obj. Identification by Distance in Eigenspace

- Objects are represented as coordinates in an \( n \)-dim. eigenspace.
- Example:
  - 3D space with points representing individual objects or a manifold representing parametric eigenspace (e.g., orientation, pose, illumination).
- Estimate parameters by finding the NN in the eigenspace.

Parametric Eigenspace

- Object identification / pose estimation
  - Find nearest neighbor in eigenspace [Murase & Nayar, UCV’95]

Eigenfaces: Key Idea

- Assume that most face images lie on a low-dimensional subspace determined by the first \( k \) (\( k<d \)) directions of maximum variance.
- Use PCA to determine the vectors \( u_1, \ldots, u_k \) that span that subspace:
  \[ x \approx \mu + w_1 u_1 + w_2 u_2 + \ldots + w_k u_k \]
- Represent each face using its “face space” coordinates \( (w_1, \ldots, w_k) \)
- Perform nearest-neighbor recognition in “face space”

Eigenfaces Example

- Training images \( x_1, \ldots, x_N \)

Eigenfaces Example 2 (Better Alignment)

- Top eigenvectors:
  \[ u_1, \ldots, u_k \]
- Mean: \( \mu \)
**SVD Properties**

- Matlab: \([u \; v] = \text{svd}(A)\)
  - where \(A = u \delta v^T\)
- \(\delta = \text{rank}(A)\)
  - Number of non-zero singular values
- \(U, V\) give us orthonormal bases for the subspaces of \(A\)
  - first \(r\) columns of \(U\): column space of \(A\)
  - last \(n-r\) columns of \(U\): left nullspace of \(A\)
  - first \(r\) columns of \(V\): row space of \(A\)
  - last \(n-r\) columns of \(V\): nullspace of \(A\)
- For \(d \leq r\), the first \(d\) columns of \(U\) provide the best \(d\)-dimensional basis for columns of \(A\) in least-squares sense
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 the corresponding eigenvectors.
• And $\sum_{i=1}^{n} \alpha_i^2 = [a_1 \ldots a_n][a_1 \ldots a_n]^T = AA^T$
• Covariance matrix $\Sigma = \frac{1}{n} \sum_{i=1}^{n} (\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T$
• So, ignoring the factor $1/n$, subtract mean image $\mu$ from each input image, create data matrix, and perform (thin) SVD on the data matrix.

Thin SVD

• Any $m$ by $n$ matrix $A$ may be factored such that $A = U \Sigma V^T$
• If $m>n$, then one can view $\Sigma$ as:
  $$\begin{bmatrix} 
  \Sigma \\
  0 
  \end{bmatrix}$$
• Where $\Sigma = \text{diag}(\sigma_1,\sigma_2,\ldots,\sigma_s)$ with $s = \min(m,n)$, and lower matrix is $(n-m)$ by $m$ of zeros.
• Alternatively, you can write:
  $$A = U \Sigma V^T$$
• In Matlab, thin SVD is:
  $$[U, S, V] = \text{svds}(A)$$

Limitations

• Global appearance method: not robust to misalignment, background variation
  - Easy fix (with considerable manual overhead)
    - Need to align the training examples
• PCA assumes that the data has a Gaussian distribution (mean $\mu$, covariance matrix $\Sigma$)
  - The shape of this dataset is not well described by its principal components
• The direction of maximum variance is not always good for classification

Topics of This Lecture

• Subspace Methods for Recognition
  - Activation
• Principal Component Analysis (PCA)
  - Derivation
  - Object recognition with PCA
  - Eigenimages/Eigenfaces
  - Limitations
• Fisher’s Linear Discriminant Analysis (LDA)
  - Derivation
  - Fisherfaces for recognition
  - Robust PCA
  - Application: robot localization
Restrictions of PCA

- PCA minimizes projection error

PCA projection
Best discriminating projection

- PCA is "unsupervised" no information on classes is used
- Discriminating information might be lost

Fischer’s Linear Discriminant Analysis (LDA)

- LDA is an enhancement to PCA
  Constructs a discriminant subspace that minimizes the scatter between images of the same class and maximizes the scatter between different class images

Mean Images

- Let \( X_1, X_2, \ldots, X_c \) be the classes in the database and let each class \( X_i \), \( i = 1, 2, \ldots, c \) have \( k \) images \( x_{ij} \), \( j = 1, 2, \ldots, k \).

- We compute the mean image \( \mu_i \) of each class \( X_i \) as:
  \[
  \mu_i = \frac{1}{k} \sum_{j=1}^{k} x_{ij}
  \]

- Now, the mean image \( \mu \) of all the classes in the database can be calculated as:
  \[
  \mu = \frac{1}{c} \sum_{i=1}^{c} \mu_i
  \]

Scatter Matrices

- We calculate the within-class scatter matrix as:
  \[
  S_w = \sum_{i=1}^{c} \sum_{x_{ij}} (x_{ij} - \mu_i)(x_{ij} - \mu_i)^T
  \]

- We calculate the between-class scatter matrix as:
  \[
  S_b = \sum_{i=1}^{c} N_i (\mu_i - \mu)(\mu_i - \mu)^T
  \]

Visualization

\( S_b = S_1 + S_2 \)

Good separation

Linear Discriminant Analysis (LDA)

- Maximize distance between classes
- Minimize distance within a class

- Criterion: \( J(w) = \frac{w^T S_b w}{w^T S_w w} \)
- \( S_b \) ... between-class scatter matrix
- \( S_w \) ... within-class scatter matrix

- Vector \( w \) is a solution of a generalized eigenvalue problem:
  \[
  S_w w = \lambda S_b w
  \]

- Classification function:
  \[
  g(x) = w^T x + w_0 \geq 0
  \]
LDA Computation

- Maximization of
  \[ J(w) = \frac{w^T S_B w}{w^T S_W w} \]
  is given by solution of generalized eigenvalue problem

\[ S_B w = \lambda S_W w \]

- Defining \( w = S_B^{-1} w \) we get

\[ S_B^{-1} S_W^{-1} \lambda w = \lambda w \]

which is a regular eigenvalue problem.

- For the c-class case we obtain (at most) c-1 projections.

Face Recognition Difficulty: Lighting

- The same person with the same facial expression, and seen from the same viewpoint, can appear dramatically different when light sources illuminate the face from different directions.

Application: Fisherfaces

- Idea:
  - Using Fisher’s linear discriminant to find class-specific linear projections that compensate for lighting/facial expression.

- Singularity problem
  - The within-class scatter is always singular for face recognition, since #training images << #pixels
  - This problem is overcome by applying PCA first

\[ W_{w} = W_{AI} W_{ps} \]

where

\[ W_{ps} = \arg \max_{w} \left| w^T S_W w \right| \]

\[ W_{AI} = \arg \max_{w} \left| w^T S_B^{-1} S_W^{-1} w \right| \]

Fisherfaces: Experiments

- Variation in lighting

Fisherfaces: Experimental Results

- Experimental results for different lighting conditions and subsets.
Fisherfaces: Experiments

- Variation in facial expression, eye wear, lighting

Fisherfaces: Experimental Results

Fisherfaces: Interpretability

- Example Fisherface for recognition “Glasses/NoGlasses”

Topics of This Lecture

- Subspace Methods for Recognition
  - Activation
- Principal Component Analysis (PCA)
  - Derivation
  - Object recognition with PCA
  - Eigenimages/Eigenfaces
  - Limitations
- Fisher’s Linear Discriminant Analysis (LDA)
  - Derivation
  - Fisherfaces for recognition
- Application: robot localization
- Robust PCA
  - Robust recognition under occlusion

Application: Mobile Robot Localization

- Environment represented by a large number of views
- Localization = recognition
Compression with PCA

Localization

Localization Results

Problems: Occlusion, Noise, Illumination

Topics of This Lecture

Robust Estimation of PCA Coefficients

- Main idea: Instead of using the standard approach
  - Select a subset of pixels
  - Find a robust solution of equations
  - Evaluate multiple hypotheses

- Hypothesize-and-test paradigm
- Competing hypotheses are subject to a selection procedure based on the MDL principle

- Standard projection for estimation of parameters gives an arbitrary error in case of noise and occlusion

- Robust PCA
- Robust recognition under occlusion
Hypothesis Selection

- Three cases
  1. One object:
     - Select best match ($c_i$)
  2. Multiple non-overlapping objects:
     - Select local maximum ($c_i$) in sampling window
  3. Multiple overlapping objects:
     - This is the only critical case.
     - Here, measurements from different objects may result in spurious estimation results.
     - This can be resolved by a model selection criterion


References and Further Reading

- Background information on PCA/LDA can be found in Chapter 22.3 of

- Important Papers (available on webpage)
  - M. Turk, A. Pentland
    Eigenfaces for Recognition
  - P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman
  - A. Leonardis, H. Bischof