Computer Vision - Lecture 9

Recognition with Global Representations II

25.11.2014

Bastian Leibe
RWTH Aachen
http://www.vision.rwth-aachen.de/

leibe@vision.rwth-aachen.de
Course Outline

- Image Processing Basics
- Segmentation & Grouping
- Recognition
  - Global Representations
  - Subspace Representations
- Object Categorization I
  - Sliding Window based Object Detection
- Local Features & Matching
- Object Categorization II
  - Part based Approaches
- 3D Reconstruction
- Motion and Tracking
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.

⇒ Fundamental paradigm shift in the 90’s
Recap: Recognition Using Histograms

- Histogram comparison

Test image

Known objects
Recap: Comparison Measures

- **Vector space interpretation**
  - Euclidean distance
  - Mahalanobis distance

- **Statistical motivation**
  - Chi-square
  - Bhattacharyya

- **Information-theoretic motivation**
  - Kullback-Leibler divergence, Jeffreys divergence

- **Histogram motivation**
  - Histogram intersection

- **Ground distance**
  - Earth Movers Distance (EMD)
Recap: Recognition Using Histograms

- **Simple algorithm**
  1. Build a set of histograms $H = \{h_i\}$ for each known object
     - More exactly, for each view of each object
  2. Build a histogram $h_t$ for the test image.
  3. Compare $h_t$ to each $h_i \in H$
     - Using a suitable comparison measure
  4. Select the object with the best matching score
     - Or reject the test image if no object is similar enough.

“Nearest-Neighbor” strategy
Generalization of the Idea

- Histograms of derivatives
  - $D_x$
  - $D_y$
  - $D_{xx}$
  - $D_{xy}$
  - $D_{yy}$
General Filter Response Histograms

- Any local descriptor (e.g. filter, filter combination) can be used to build a histogram.

- **Examples:**
  - Gradient magnitude
    \[ Mag = \sqrt{D_x^2 + D_y^2} \]
  - Gradient direction
    \[ Dir = \arctan \frac{D_y}{D_x} \]
  - Laplacian
    \[ Lap = D_{xx} + D_{yy} \]
Multidimensional Representations

- Combination of several descriptors
  - Each descriptor is applied to the whole image.
  - Corresponding pixel values are combined into one feature vector.
  - Feature vectors are collected in multidimensional histogram.

\[ D_x, D_y, \text{Lap} \]

\[
\begin{array}{c}
1.22 \\
-0.39 \\
2.78
\end{array}
\]
Multidimensional Histograms

- Examples

[B. Leibe]

[Schiele & Crowley, 2000]
Multidimensional Representations

• Useful simple combinations
  - $D_x - D_y$
    - Rotation-variant
      - Descriptor changes when image is rotated.
      - Useful for recognizing oriented structures (e.g. vertical lines)
  - Mag-Lap
    - Rotation-invariant
      - Descriptor does not change when image is rotated.
      - Can be used to recognize rotated objects.
      - Less discriminant than rotation-variant descriptor.
Special Case: Multiscale Representations

- Combination of several scales
  - Descriptors are computed at different scales.
  - Each scale captures different information about the object.
  - Size of the support region grows with increasing $\sigma$.
  - Feature vectors capture both local details and larger-scale structures.

\[
\begin{array}{c|c|c|c}
\sigma = 2.0 & 1.22 & -0.39 & 2.78 \\
\sigma = 4.0 & & & \\
\sigma = 8.0 & & & \\
\end{array}
\]
Generalization: Filter Banks

- What filters to put in the bank?
  - Typically we want a combination of scales and orientations, different types of patterns.

Matlab code available for these examples: [http://www.robots.ox.ac.uk/~vgg/research/texclass/filters.html](http://www.robots.ox.ac.uk/~vgg/research/texclass/filters.html)
Example Application of a Filter Bank

Filter bank of 8 filters

Input image

8 response images: magnitude of filtered outputs, per filter

Slide credit: Kristen Grauman
Extension: Colored Derivatives

- **YC₁C₂ color space**

\[
\begin{pmatrix}
Y \\
C_1 \\
C_2
\end{pmatrix} = \begin{pmatrix}
\frac{g_r}{2} & \frac{g_g}{2} & \frac{g_b}{2} \\
\frac{3g_g}{2} & -\frac{3g_r}{2} & 0 \\
\frac{gbg_r}{g_r^2 + g_g^2} & \frac{gbg_g}{g_r^2 + g_g^2} & -1
\end{pmatrix}
\begin{pmatrix}
R \\
G \\
B
\end{pmatrix}
\]

- **Color-opponent space**
  - Inspired by models of the human visual system
  - Y \(\equiv\) intensity
  - C₁ \(\equiv\) red-green
  - C₂ \(\equiv\) blue-yellow

---

[Hall & Crowley, 2000]
Extension: Colored Derivatives

- Generalization: derivatives along
  - Y axis $\rightarrow$ intensity differences
  - $C_1$ axis $\rightarrow$ red-green differences
  - $C_2$ axis $\rightarrow$ blue-yellow differences

- Feature vector is rotated such that $D_y = 0$
  - Rotation-invariant descriptor

[Hall & Crowley, 2000]
Summary: Multidimensional Representations

- **Pros**
  - Work very well for recognition.
  - Usually, simple combinations are sufficient (e.g. $D_x$-$D_y$, Mag-Lap)
  - But multiple scales are very important!
  - Generalization: filter banks

- **Cons**
  - High-dimensional histograms $\Rightarrow$ lots of storage space
  - Global representation $\Rightarrow$ not robust to occlusion
You’re Now Ready for First Applications…

- Binary Segmentation
- Line detection
- Circle detection
- Skin color detection
- Moment descriptors
- Histogram based recognition

Image Source: http://www.flickr.com/photos/angelsk/2806412807/
Topics of This Lecture

• **Subspace Methods for Recognition**
  - Motivation

• **Principal Component Analysis (PCA)**
  - Derivation
  - Object recognition with PCA
  - Eigenimages/Eigenfaces
  - Limitations

• **Discussion: Global representations for recognition**
  - Vectors of pixel intensities
  - Histograms
  - Localized Histograms

• **Application: Image completion**
Representations for Recognition

• Global object representations
  - We’ve seen histograms as one example
  - What could be other suitable representations?

• 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

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

Slide adapted from Ales Leonardis
Subspace Methods

- Today’s topic: PCA

Slide credit: Ales Leonardis
Topics of This Lecture

• Subspace Methods for Recognition
  ➢ Motivation

• Principal Component Analysis (PCA)
  ➢ Derivation
  ➢ Object recognition with PCA
  ➢ Eigenimages/Eigenfaces
  ➢ Limitations

• Discussion: Global representations for recognition
  ➢ Vectors of pixel intensities
  ➢ Histograms
  ➢ Localized Histograms

• Application: Image completion
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?
Principal Component Analysis

- Direction that maximizes the variance of the projected data:

\[
\text{var}(u) = \frac{1}{N} \sum_{i=1}^{N} u^T (x_i - \mu)(u^T (x_i - \mu))^T
\]

- Projection of data point

\[
= \frac{1}{N} u^T \left[ \sum_{i=1}^{N} (x_i - \mu)(x_i - \mu)^T \right] u
\]

- Covariance matrix of data

\[
= \frac{1}{N} u^T \Sigma u
\]

- The direction that maximizes the variance is the eigenvector associated with the largest eigenvalue of \( \Sigma \).
Remember: Fitting a Gaussian

- Mean and covariance matrix of data define a Gaussian model
Interpretation of PCA

- Compute eigenvectors of covariance $\Sigma$.
  - Eigenvectors: main directions
  - Eigenvalues: variances along eigenvector

- Result: coordinate transform to best represent the variance of the data
Interpretation of PCA

• Now, suppose we want to represent the data using just a single dimension.
  - i.e., project it onto a single axis
  - What would be the best choice for this axis?
Interpretation of PCA

• Now, suppose we want to represent the data using just a single dimension.
  - I.e., project it onto a single axis
  - What would be the best choice for this axis?

• The first eigenvector gives us the best reconstruction.
  - Direction that retains most of 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:

$$\sum_{j=1}^{N} \lambda_j - \sum_{j=1}^{m} \lambda_j = \sum_{j=m+1}^{N} \lambda_j$$

where $\lambda_j$ are the eigenvalues

• 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
  \[ y = Ux \]

- From $y \in \mathbb{R}^m$, the reconstruction of the point is $U^Ty$

- The error of the reconstruction is
  \[ \| x - U^TUx \| \]
Example: Object Representation
Principal Component Analysis

Get a compact representation by keeping only the first $k$ eigenvectors!

$$= w_1 + w_2 + w_3$$
Object Detection by Distance TO Eigenspace

- Is an image window $\omega$ likely to contain a learned object?
  - 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
Eigenfaces: Key Idea

- Assume that most face images lie on a low-dimensional subspace determined by the first $k$ directions of maximum variance (where $k < d$).

- Use PCA to determine the vectors $u_1, ... u_k$ that span that subspace:
  \[ x \approx \mu + w_1u_1 + w_2u_2 + ... + w_ku_k \]

- Represent each face using its “face space” coordinates $(w_1, ... w_k)$

- Perform nearest-neighbor recognition in “face space”


Slide credit: Svetlana Lazebnik
Eigenfaces Example

- Training images: $x_1, \ldots, x_N$
Eigenfaces Example

Top eigenvectors: $u_1, \ldots, u_k$

Mean: $\mu$

Slide credit: Svetlana Lazebnik
Eigenface Example 2 (Better Alignment)
Eigenfaces Example

- Face $x$ in “face space” coordinates:

$$x \rightarrow [u_1^T(x - \mu), \ldots, u_k^T(x - \mu)]$$

$$= \omega_1, \ldots, \omega_k$$
Eigenfaces Example

- Face $x$ in “face space” coordinates:

$$x \rightarrow [u_1^T(x - \mu), \ldots, u_k^T(x - \mu)]$$

$$= \omega_1, \ldots, \omega_k$$

- Reconstruction:

$$x = \mu + w_1u_1 + w_2u_2 + w_3u_3 + w_4u_4 + \ldots$$

Slide credit: Svetlana Lazebnik
Recognition with Eigenspaces

- Process labeled training images:
  - Find mean $\mu$ and covariance matrix $\Sigma$
  - Find $k$ principal components (eigenvectors of $\Sigma$) $u_1, \ldots, u_k$
  - Project each training image $x_i$ onto subspace spanned by principal components:
    \[(w_{i1}, \ldots, w_{ik}) = (u_1^T(x_i - \mu), \ldots, u_k^T(x_i - \mu))\]

- Given novel image $x$:
  - Project onto subspace:
    \[(w_1, \ldots, w_k) = (u_1^T(x - \mu), \ldots, u_k^T(x - \mu))\]
  - Optional: check reconstruction error $x - \hat{x}$ to determine whether image is really a face
  - Classify as closest training face in $k$-dimensional subspace
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.

Slide credit: Ales Leonardis
Parametric Eigenspace

- Object identification / pose estimation
  - Find nearest neighbor in eigenspace [Murase & Nayar, IJCV’95]
Applications: Recognition, Pose Estimation

H. Murase and S. Nayar, Visual learning and recognition of 3-d objects from appearance, IJCV 1995
Applications: Visual Inspection


B. Leibe
Important Footnote

- Don’t really implement PCA this way!
  - Why?

1. How big is $\Sigma$?
   - $n \times n$, where $n$ is the number of pixels in an image!
   - However, we only have $m$ training examples, typically $m \ll n$.
     $\Rightarrow \Sigma$ will at most have rank $m$!

2. You only need the first $k$ eigenvectors
Singular Value Decomposition (SVD)

- Any $m \times n$ matrix $A$ may be factored such that
  \[ A = U\Sigma V^T \]
  \[ [m \times n] = [m \times m][m \times n][n \times n] \]
- $U$: $m \times m$, orthogonal matrix
  - Columns of $U$ are the eigenvectors of $AA^T$
- $V$: $n \times n$, orthogonal matrix
  - Columns are the eigenvectors of $A^TA$
- $\Sigma$: $m \times n$, diagonal with non-negative entries ($\sigma_1, \sigma_2, \ldots, \sigma_s$) with $s = \min(m, n)$ are called the singular values.
  - Singular values are the square roots of the eigenvalues of both $AA^T$ and $A^TA$. Columns of $U$ are corresponding eigenvectors!
  - Result of SVD algorithm: $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_s$
SVD Properties

- Matlab: \([u \ s \ v] = \text{svd}(A)\)
  - where \(A = u*s*v'\)

- \(r = \text{rank}(A)\)
  - 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 \(m-r\) 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

Slide credit: Peter Belhumeur
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
  \[
  \sum_{i=1}^{n} a_i a_i^T = [a_1 \ \ldots \ a_n][a_1 \ \ldots \ a_n]^T = AA^T
  \]

- Covariance matrix
  \[
  \Sigma = \frac{1}{n} \sum_{i=1}^{n} (\bar{x}_i - \mu)(\bar{x}_i - \mu)^T
  \]

- So, ignoring the factor $1/n$, subtract mean image $\mu$ from each input image, create data matrix $A = (\bar{x}_i - \mu)$, and perform (thin) SVD on the data matrix.
Limitations

- Global appearance method: not robust to misalignment, background variation

- Easy fix (with considerable manual overhead)
  - Need to align the training examples
Limitations

- 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

Slide credit: Svetlana Lazebnik
Limitations

• The direction of maximum variance is not always good for classification
Topics of This Lecture

• Subspace Methods for Recognition
  ➢ Motivation

• Principal Component Analysis (PCA)
  ➢ Derivation
  ➢ Object recognition with PCA
  ➢ Eigenimages/Eigenfaces
  ➢ Limitations

• Discussion: Global representations for recognition
  ➢ Vectors of pixel intensities
  ➢ Histograms
  ➢ Localized Histograms

• Application: Image completion
Feature Extraction: Global Appearance

- Simple holistic descriptions of image content
  - Vector of pixel intensities
Eigenfaces: Global Appearance Description

This can also be applied in a sliding-window framework...

Training images

Mean

Eigenvectors computed from covariance matrix

Generate low-dimensional representation of appearance with a linear subspace.

Project new images to “face space”.

Recognition via nearest neighbors in face space

\[ x \approx \text{Mean} + \sum_{i=1}^{k} w_i \]

[Turk & Pentland, 1991]
Feature Extraction: Global Appearance

- Simple holistic descriptions of image content
  - Vector of pixel intensities
  - Pixel based representations sensitive to small shifts!
Feature Extraction: Global Appearance

- Simple holistic descriptions of image content
  - Vector of pixel intensities
  - Grayscale / color histograms

⇒ Color or grayscale-based appearance description can be sensitive to illumination and intra-class appearance variation!
Gradient-based Representations

• Better: Edges, contours, and (oriented) intensity gradients
Matching Edge Templates

- Example: Chamfer matching

At each window position, compute average min distance between points on template (T) and input (I).

\[ D_{\text{chamfer}}(T, I) \equiv \frac{1}{|T|} \sum_{t \in T} d_I(t) \]
Gradient-based Representations

• Improved discriminance: localized gradients

• Summarize local distribution of gradients with histogram
  - Locally orderless: offers invariance to small shifts and rotations
  - Contrast-normalization: try to correct for variable illumination

Slide credit: Kristen Grauman
Gradient-based Representations: Histograms of Oriented Gradients (HOG)

Map each grid cell in the input window to a histogram counting the gradients per orientation.

Code available: http://pascal.inrialpes.fr/soft/olt/

[Dalal & Triggs, CVPR 2005]
References and Further Reading

- Background information on PCA 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