Beyond Bags of features
Spatial information & Shape models

Jana Kosecka

Many slides adapted from S. Lazebnik, Fei-Fei Li, Rob Fergus, and Antonio Torralba

Detection, recognition (so far ...)
- Bags of features models, codebooks made based on appearance
- No spatial relationships between local features
- Incorporating spatial information
- Edge based representations
- Distance Transform, Chamfer matching
- Generalized Hough Transform
- Combinations of edge based and patch based

Adding spatial information
- Forming vocabularies from pairs of nearby features — “doublets” or “bigrams”
- Computing bags of features on sub-windows of the whole image
- Using codebooks to vote for object position
- Generative part-based models

Spatial pyramid representation
- Extension of a bag of features
- Locally orderless representation at several levels of resolution

Lazebnik, Schmid & Ponce (CVPR 2006)
Spatial pyramid representation
- Extension of a bag of features
- Locally orderless representation at several levels of resolution

Lazebnik, Schmid & Ponce (CVPR 2006)

Scene category dataset

Multi-class classification results
(100 training images per class)

<table>
<thead>
<tr>
<th>Level</th>
<th>Weak features (vocabulary size: 16)</th>
<th>Strong features (vocabulary size: 200)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 (1 × 1)</td>
<td>45.3 ±0.5</td>
<td>72.2 ±0.6</td>
</tr>
<tr>
<td>1 (2 × 2)</td>
<td>53.6 ±0.1</td>
<td>56.2 ±0.6</td>
</tr>
<tr>
<td>2 (4 × 4)</td>
<td>61.7 ±0.6</td>
<td>64.7 ±0.7</td>
</tr>
<tr>
<td>3 (8 × 8)</td>
<td>63.3 ±0.8</td>
<td><strong>66.8 ±0.6</strong></td>
</tr>
</tbody>
</table>

Caltech101 dataset


Multi-class classification results (30 training images per class)

<table>
<thead>
<tr>
<th>Level</th>
<th>Weak features (16)</th>
<th>Strong features (200)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>15.5 ±0.9</td>
<td>41.3 ±1.2</td>
</tr>
<tr>
<td>1</td>
<td>31.4 ±1.2</td>
<td>32.8 ±1.3</td>
</tr>
<tr>
<td>2</td>
<td>47.2 ±1.1</td>
<td>49.3 ±1.4</td>
</tr>
<tr>
<td>3</td>
<td>52.2 ±0.8</td>
<td><strong>54.0 ±1.1</strong></td>
</tr>
</tbody>
</table>
Edge Based Representations - Shape

- Edge based representations
- Distance Transform, Chamfer matching
- Generalized Hough Transform
- Combinations of edge based and patch based

Haussdorf Distance Matching

- Given and edge template
- Find it in the image

Haussdorf distance matching

- Let M be an \( nxn \) binary template and N an \( nxn \) binary image we want to compare to that template
- \( H(M,N) = \max(h(M, N), h(N, M)) \) where
  \[
  h(A, B) = \max_{A \in A} \min_{B \in B} |A - B|
  \]
  \( || \) is a distance function like the Euclidean distance function
  - For each pixel from B find the closest from A and pick the largest of those distances
  - \( h(A, B) \) is called the directed Hausdorff distance.
  - ranks each point in A based on closeness to a point in B
  - most mis-matched point is measure of match
  - if \( h(A, B) = e \), then all points in A must be within distance \( e \) of B.
  - generally, \( h(A, B) \neq h(B, A) \)
  - easy to compute Hausdorff distances from distance transform

Distance Transform

- Given:
  - binary image, \( B_e \) of edge and local feature locations
  - binary "edge template", \( T_e \) of shape we want to match
- Let D be an array in registration with B such that \( D(i,j) \) is the distance to the nearest "1" in B.
  - this array is called the distance transform of B (binary image)
Distance Transform

- Use of distance transform for template matching
- Goal: Find placement of template T in D that minimizes the sum, M, of the distance transform multiplied by the pixel values in T
  - If T is an exact match to B at location (i,j) then M(i,j) = 0
  - I.e. all non-zero pixels of T will have distance 0
  - If the edges in B are slightly displaced from their ideal locations in T, we still get a good match using the distance transform technique

Computing the distance transform

- Two pass sequential algorithm
- Initialize: set $D(i,j) = 0$ where $B(i,j) = 1$, else set $D(i,j) = \infty$
- Forward pass
  - $D(i,j) = \min\{ D(i-1,j-1) + 1, D(i-1,j) + 1, D(i-1,j+1) + 1, D(i,j-1) + 1, D(i,j) \}$
- Backward pass
  - $D(i,j) = \min\{ D(i,j+1) + 1, D(i+1,j-1) + 1, D(i+1,j) + 1, D(i+1,j+1) + 1, D(i,j) \}$

### Example

```
\begin{array}{ccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
2 & 2 & 2 & 2 & 2 & 2 & 2 \\
3 & 3 & 3 & 3 & 3 & 3 & 3 \\
4 & 4 & 4 & 4 & 4 & 4 & 4 \\
5 & 5 & 5 & 5 & 5 & 5 & 5 \\
\end{array}
```
Distance transform example

D(i,j) = min[D(i,j), D(i,j) + 1, D(i+1,j) + 1, D(i+1,j+1) + 1]

Chamfer matching

- Chamfer matching is convolution of a binary edge template with the distance transform
- for any placement of the template over the image, it sums up the distance transform values for all pixels that are "1's" (edges) in the template
- if, at some position in the image, all of the edges in the template coincide with edges in the image (which are the points at which the distance transform is zero), then we have a perfect match with a match score of 0.

Hausdorff distance matching

- Let M be an nxn binary template and N an nxn binary image we want to compare to that template
- \( h(M, N) = \max(h(M, N), h(N, M)) \) where \( || \cdot || \) is a distance function like the Euclidean distance function
- \( h(A, B) \) is called the directed Hausdorff distance.
- \( h(A, B) \) ranks each point in A based on closeness to a point in B
- most mis-matched point is measure of match
- if \( h(A, B) = e \), then all points in A must be within distance e of B.
- generally, \( h(A, B) \) is not equal to \( h(B, A) \)
- easy to compute Hausdorff distances from distance transform
Generalized Hough Transform

- Representation of arbitrary shapes
- Generalized Hough Transform
- Special case: object of fixed 2D orientation and size
  - Idea: pick reference point on the shape: i.e. middle
  - Build R-table (look-up table) where each boundary point with normal \( \phi \) store the distance and angle where it occurs normal
  - Detection: given boundary points and gradients – vote for object centers

\[ x_c = x + r \cos(\alpha) \]
\[ y_c = y + r \sin(\alpha) \]

Possible locations of the object center are the extrema in the accumulator array.
Extensions for objects in varying orientations and scales

Other generalizations

- Representation of arbitrary shapes
- Generalized Hough Transform
- Match patterns of linear and curvilinear features against images from which such features have been detected
- Impose a hierarchical structure on \( M \), and match pieces and compositions of pieces.
  - at lowest level one finds possible matches to small pieces of \( M \)
  - a second GHT algorithm can now find combinations of pieces that satisfy other spatial constraints.

Summary 2-D object recognition

- Binary vision systems
  - segmentation by thresholding and connected component analysis
  - object modeling using statistical techniques
    - means and variances of global object features such as area, perimeter, etc.
    - recognition using statistical recognition techniques
      - k-nearest neighbors
      - Bayesian recognition
  - Drawbacks
    - touching objects
    - occluded objects
    - weak segmentation techniques

- Grey level vision systems
  - (optional) segmentation by edge detection
  - object modeling by templates
    - gray level region templates
    - edge templates (binary)
  - recognition using correlation
    - brute force image correlation
    - speedup methods
    - Hough transform methods
    - Chamfer matching
  - Drawbacks
    - computational complexity
    - to support rotations and scaling of templates
Hough Transforms for line matching

- Let \( L = \{L_1, ..., L_n\} \) be the set of line segments which define \( M \)
- Let \( L' = \{L'_1, ..., L'_m\} \) be the set of observed line segments from \( N \)
- Define \( L_i - L_j \) as follows:
  - If \( L_j \) is a subsegment of \( L_i \), then \( L_i - L_j = l_j \), where \( l_j \) is the length of \( L_j \)
  - otherwise \( L_i - L_j = 0 \)
- Let \( F \) be a set of transformations that map lines to lines
- Given \( F \), \( L \) and \( L' \), find \( f \) in \( F \) that maximizes:

\[
\nu(f) = \sum_{i,j} [u - f(l_{ij})]
\]

Example - translation only

- Which translations get incremented
  - \( \alpha\)-a: \( (0,6), (1,6), (2,6), (3,6) \) incremented by 2
  - \( \alpha\)-b: none
  - \( \alpha\)-c: \( (2,0), (2,1) \) incremented by 2

Fast template matching

- Simulated annealing approach
  - Let \( T_{\theta,s} \) be a rotated and scaled version of \( T \)
  - For a random \( \theta \) and \( s \), and a random \( (i,j) \) match \( T_{\theta,s} \) at position \( (i,j) \) of \( I \)
    - Now, randomly perturb \( \theta, s, i, j \) by perturbations whose magnitudes will be reduced in subsequent iterations of the algorithm to obtain \( \theta', s', i', j' \).
    - Match \( T_{\theta',s'} \) at position \( (i',j') \). If the match is better, “move” to that position in the search space. If the match is worse, move with some probability to that position anyway!
    - Iterate using smaller perturbations, and smaller probabilities of moving to worse locations
      - the rate at which the probability of taking “bad” moves decreases is called the “cooling schedule” of the process.
  - This has also been demonstrated with deformation parameters that mimic projection effects for planar patterns.

Implicit shape models

- Combining the edge based GHT style voting with appearance codebooks
- Visual codebook is used to index votes for object position

\[ \text{training image annotated with object localization info} \]
\[ \text{visual codeword with displacement vectors} \]

B. Leibe, A. Leonardis, and B. Schiele,
Combined Object Categorization and Segmentation with an Implicit Shape Model,
ECCV Workshop on Statistical Learning in Computer Vision 2004
Implicit shape models

- Visual codebook is used to index votes for object position

Idea Implicit Shape Model

- Faces rectangular templates – detection windows
- Does not generalize to more complex object with different shapes
- How to combine patch based – appearance based representations to incorporate notion of shape
- Combined Object Categorization and Segmentation with an Implicit Shape Model. Bastian Leibe, Alex Leonardis, and Bernt Schiele. In ECCV’04.

Initial Recognition Approach

- First Step: Generate hypotheses from local features
- Training: Agglomerative Clustering

- How to decide when to merge two clusters
- Average NCC of patches

\[ \text{NCC}(P, Q) = \frac{\sum (p_i - \bar{p})(q_i - \bar{q})}{\sqrt{\sum (p_i - \bar{p})^2 \sum (q_i - \bar{q})^2}} \]

- Lowe’s DoG Detector
  - Resize to 25 x 25
  - Find codebook patches
  - Learn Spatial Distribution

- Initial Recognition Approach
  - Codebook words - spatial information is lost
  - For each codebook entry store all positions it was activated in relative to object center (positions parametrized by \( r \) and \( \theta \))
  - Parts vote for object center

- Refine Hypothesis (uniform sampling)
- Backprojected Hypothesis
- Backprojection of Maximum

Voting Space (continuous)

30
Pedestrian Detection in Crowded Scenes

1. Interleaved Object Categorization and Segmentation, BMVC’03

Pedestrian Detection
- Many applications
- Large variation in shape, appearance
- Need to combine different representations
- Basic Premise: "[Such a] problem is too difficult for any type of feature or model alone"
- Probabilistic bottom up, top down segmentation

Theme of the Paper
- Open Question: How would you do pedestrian detection/segmentation?
- Solution: integrate as many cues as possible from many sources

Datasets
- Training Set: 35 people walking parallel to the image plane
- Testing Set (Much harder!): 209 images of 595 annotated pedestrians
Theme of the Paper

Initial Recognition Approach
- First Step: Generate hypotheses from local features (Intrinsic Shape Models)
- Testing:
  - Initial Hypothesis: Overall

Caveat: it leads to another set of problems

ISM doesn’t know a person doesn’t have three legs!

Global Cues are needed

Or four legs and three arms
ISM doesn’t know a person doesn’t have three legs!
Assimilation of Global Cues

- Distance Transform, Chamfer Matching

\[ D_{\text{chamfer}}(T, I) = \sqrt{\sum_{ij} \min|T(j)|} \]

get feature image by an edge detector
get DT image by computing distance to nearest feature point
Chamfer Distance between template and DT image

Assimilation of Global Cues (Attempt 1)

- Distance Transform, Chamfer Matching

- Use scale estimate to cut out surrounding region
- Apply Canny detector and compute DT
- Chamfer distance based matching

Results

Generative Part Based Models

Many slides adapted from Fei-Fei Li, Rob Fergus, and Antonio Torralba
Generative part-based models

R. Fergus, P. Perona and A. Zisserman,
Object Class Recognition by Unsupervised Scale-Invariant Learning, CVPR 2003

Probabilistic model

\[ P(\text{image} | \text{obj ecch}) = P(\text{appearance}, \text{shape} | \text{obj ecch}) \]

Part descriptors
Part locations

Candidate parts

Probabilistic model

\[ P(\text{image} | \text{obj ecch}) = P(\text{appearance}, \text{shape} | \text{obj ecch}) \]

\[ = \max_h P(\text{appearance} | h, \text{obj ecch}) P(\text{shape} | h, \text{obj ecch}) P(h | \text{obj ecch}) \]

h: assignment of features to parts
Probabilistic model

\[ P(\text{image} \mid \text{obj ech}) = P(\text{appearance}, \text{shape} \mid \text{obj ech}) \]

\[ = \max_h P(\text{appearance} \mid h, \text{obj ech}) \cdot P(\text{shape} \mid h, \text{obj ech}) \cdot p(h \mid \text{obj ech}) \]

Distribution over patch descriptors

High-dimensional appearance space

Probabilistic model

\[ P(\text{image} \mid \text{obj ech}) = P(\text{appearance}, \text{shape} \mid \text{obj ech}) \]

\[ = \max_h P(\text{appearance} \mid h, \text{obj ech}) \cdot P(\text{shape} \mid h, \text{obj ech}) \cdot p(h \mid \text{obj ech}) \]

Distribution over joint part positions

2D image space

How to model location?

- Explicit: Probability density functions
- Implicit: Voting scheme
- Invariance
  - Translation
  - Scaling
  - Similarity/affine
  - Viewpoint

Explicit shape model

- Probability densities
  - Continuous (Gaussians)
  - Analogy with springs
- Parameters of model, \( \mu \) and \( \Sigma \)
  - Independence corresponds to zeros in \( \Sigma \)

\[
\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \quad \Sigma = \begin{pmatrix} 
\sigma_{11} & \sigma_{12} \\ 
\sigma_{21} & \sigma_{22} 
\end{pmatrix}
\]

\[
\sigma_{11} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \end{pmatrix}, \quad \sigma_{22} = \begin{pmatrix} 11 & 12 & 13 & 14 & 15 \\ 16 & 17 & 18 & 19 & 20 \end{pmatrix}
\]
Shape

- Shape is “what remains after differences due to translation, rotation, and scale have been factored out”. [Kendall84]
  $$p(x_1, \ldots, x_P; y_1, \ldots, y_P) = p_{Y|x}(x, y) p_{X|\theta}(x_1, \ldots, x_P; y_1, \ldots, y_P)$$

- Statistical theory of shape [Kendall, Bookstein, Mardia & Dryden]

Euclidean & Affine Shape

- Translation, rotation and scaling $\Rightarrow$ Euclidean Shape
- Removal of camera foreshortenings $\Rightarrow$ Affine Shape

Assume Gaussian density in figure space

What is the probability density for the shape variables in each of the different spaces?

Results: Faces

Face shape model

Patch appearance model

Recognition results

Results: Motorbikes and airplanes
Summary: Adding spatial information

- Doublet vocabularies
  - Pro: takes co-occurrences into account, some geometric invariance is preserved
  - Cons: too many doublet probabilities to estimate
- Spatial pyramids
  - Pro: simple extension of a bag of features, works very well
  - Cons: no geometric invariance, no object localization
- Implicit shape models
  - Pro: can localize object, maintain translation and possibly scale invariance
  - Cons: need supervised training data (known object positions and possibly segmentation masks)
- Generative part-based models
  - Pro: very nice conceptually, can be learned from unsegmented images
  - Cons: combinatorial hypothesis search problem