part 5: categorization
Main problem with bag-of-words

- All have equal probability for bag-of-words methods
- Location information is important
Constellation Models: Parts and Structure
Representation

- Model has two components
  - parts
    - (2D image fragments)
  - structure
    - (configuration of parts)

- Issues:
  - How to model location
  - How to represent appearance
  - Sparse or dense (pixels or regions)
  - How to handle occlusion/clutter

Figure from [Fischler73]
Example scheme

- Model shape using Gaussian distribution on location between parts
- Model appearance as pixel templates
Sparse representation

+ Computationally tractable (10^5 pixels $\rightarrow$ 10^1 -- 10^2 parts)
+ Generative representation of class
+ Avoid modeling global variability
+ Success in specific object recognition

- Throw away most image information
- Parts need to be distinctive to separate from other classes
The correspondence problem

- Model with $P$ parts
- Image with $N$ possible locations for each part

$N^P$ combinations!!!
Different graph structures

- Fully connected
  \[ O(N^6) \]

- Star structure
  \[ O(N^2) \]

- Tree structure
  \[ O(N^2) \]

• Sparser graphs cannot capture all interactions between parts
Spatial Models Considered Here

Fully connected shape model

```
X_1
X_6
X_5
X_4
X_3
X_2
```

e.g. Constellation Model
Parts fully connected
Recognition complexity: O(N^P)
Method: Exhaustive search

“Star” shape model

e.g. ISM
Parts mutually independent
Recognition complexity: O(NP)
Method: Gen. Hough Transform
How to model location?

- Explicit: Probability density functions
- Implicit: Voting scheme

- Invariance
  - Translation
  - Scaling
  - Similarity/affine
  - Viewpoint
Learning situations

- Varying levels of supervision
  - Unsupervised
  - Image labels
  - Object centroid/bounding box
  - Segmented object
  - Manual correspondence (typically sub-optimal)

- Generative models naturally incorporate labelling information (or lack of it)

- Discriminative schemes require labels for all data points
Constellation Model

Gaussian shape pdf

Gaussian part appearance pdf

Gaussian relative scale pdf

Clutter model

Uniform shape pdf

Gaussian appearance pdf

Uniform relative scale pdf

Slide credit: Grauman & Leibe
Learning using EM

- Task: Estimation of model parameters

- Chicken and Egg type problem, since we initially know neither:
  - Model parameters
  - Assignment of regions to parts

- Let the assignments be a hidden variable and use EM algorithm to learn them and the model parameters
Learning procedure

- Find regions & their location & appearance
- Initialize model parameters
- Use EM and iterate to convergence:
  - E-step: Compute assignments for which regions belong to which part
  - M-step: Update model parameters
- Trying to maximize likelihood – consistency in shape & appearance
Example scheme, using EM for maximum likelihood learning

1. Current estimate of $\theta$
2. Assign probabilities to constellations

3. Use probabilities as weights to re-estimate parameters. Example: $\mu$

\[
\text{Large P} \times \text{Small P} \times \cdots = \text{new estimate of } \mu
\]
Example: Motorbikes

Slide credit: Grauman & Leibe
Example: Motorbikes (2)

Visual Object Recognition - 67777

Slide credit: Grauman & Leibe
Example: Spotted Cats

<table>
<thead>
<tr>
<th>Part 1 - Det:8e-22</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image of spotted cats" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part 2 - Det:2e-22</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image2.png" alt="Image of spotted cats" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part 3 - Det:5e-22</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image3.png" alt="Image of spotted cats" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part 4 - Det:2e-27</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image4.png" alt="Image of spotted cats" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part 5 - Det:1e-22</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image5.png" alt="Image of spotted cats" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part 6 - Det:3e-21</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image6.png" alt="Image of spotted cats" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Background - Det:2e-18</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image7.png" alt="Image of spotted cats" /></td>
</tr>
</tbody>
</table>
Implicit Shape Model (ISM)

- Basic ideas
  - Learn an appearance codebook
  - Learn a star-topology structural model
    (Features are considered independent given obj. center)

- Algorithm: probabilistic Gen. Hough Transform
**Codebook Representation**

- Extraction of local object features
  - Interest Points (e.g. Harris detector)
  - Sparse representation of the object appearance

- Collect features from whole training set

- Example:

Slide credit: Grauman & Leibe
Appearance Codebook

- Visual similarity preserved
- Wheel parts, window corners, fenders, ...
- Store cluster centers as *Appearance Codebook*
Gen. Hough Transform with Local Features

- For every feature, store possible “occurrences”

- Object identity
- Pose
- Relative position

For new image, let the matched features vote for possible object positions
Implicit Shape Model - Recognition

Interest Points

Matched Codebook Entries

Probabilistic Voting

Visual Object Recognition - 67777

Slide credit: Grauman & Leibe
Implicit Shape Model - Recognition

Interest Points

Matched Codebook Entries

Probabilistic Voting

Backprojected Hypotheses

Backprojection of Maxima

3D Voting Space (continuous)
Discussion: Constellation Model

- **Advantages**
  - Works well for many different object categories
  - Can adapt well to categories where
    - Shape is more important
    - Appearance is more important
  - Everything is learned from training data
  - Weakly-supervised training possible

- **Disadvantages**
  - Model contains many parameters that need to be estimated
  - Cost increases exponentially with increasing number of parameters

⇒ Fully connected model restricted to small number of parts.

Slide credit: Grauman & Leibe
Summary

- Main issues in categorization:
  - **Representation** - how to represent an object category
  - **Learning** - how to form the classifier, given training data
  - **Recognition** - how the classifier is to be used on novel data

- Methods reviewed here
  - Bag of features
  - Parts and structure (via generative models)
  - Discriminative methods