Beyond bags of features: Adding spatial information

Many slides adapted from Fei-Fei Li, Rob Fergus, and Antonio Torralba
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
From single features to “doublets”

1. Run pLSA on a regular visual vocabulary
2. Identify a small number of top visual words for each topic
3. Form a “doublet” vocabulary from these top visual words
4. Run pLSA again on the augmented vocabulary

J. Sivic, B. Russell, A. Efros, A. Zisserman, B. Freeman, Discovering Objects and their Location in Images, ICCV 2005
From single features to “doublets”

Ground truth  All features  “Face” features initially found by pLSA

One doublet  Another doublet  “Face” doublets

J. Sivic, B. Russell, A. Efros, A. Zisserman, B. Freeman, *Discovering Objects and their Location in Images*, ICCV 2005
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)
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

![Image of scene categories: office, kitchen, living room, bedroom, store, industrial, tall building, inside city, street, highway, coast, open country, mountain, forest, suburb]

## 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></td>
<td>Single-level</td>
<td>Pyramid</td>
</tr>
<tr>
<td>0 (1 × 1)</td>
<td>45.3 ± 0.5</td>
<td></td>
</tr>
<tr>
<td>1 (2 × 2)</td>
<td>53.6 ± 0.3</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></td>
<td>Single-level</td>
<td>Pyramid</td>
</tr>
<tr>
<td>0</td>
<td>15.5 ±0.9</td>
<td>32.8 ±1.3</td>
</tr>
<tr>
<td>1</td>
<td>31.4 ±1.2</td>
<td>49.3 ±1.4</td>
</tr>
<tr>
<td>2</td>
<td>47.2 ±1.1</td>
<td>54.0 ±1.1</td>
</tr>
<tr>
<td>3</td>
<td>52.2 ±0.8</td>
<td></td>
</tr>
</tbody>
</table>
Implicit shape models

- Visual codebook is used to index votes for object position

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

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: Training

1. Build codebook of patches around extracted interest points using clustering
Implicit shape models: Training

1. Build codebook of patches around extracted interest points using clustering
2. Map the patch around each interest point to closest codebook entry
Implicit shape models: Training

1. Build codebook of patches around extracted interest points using clustering
2. Map the patch around each interest point to closest codebook entry
3. For each codebook entry, store all positions it was found, relative to object center
Implicit shape models: Testing

1. Given test image, extract patches, match to codebook entry
2. Cast votes for possible positions of object center
3. Search for maxima in voting space
4. Extract weighted segmentation mask based on stored masks for the codebook occurrences
Generative part-based models

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

\[ P(image \mid object) = P(appearance, shape \mid object) \]

- **Part descriptors**
- **Part locations**

Candidate parts
Probabilistic model

\[ P(\text{image} \mid \text{object}) = P(\text{appearance, shape} \mid \text{object}) \]
Probabilistic model

\[ P(\text{image} \mid \text{object}) = P(\text{appearance}, \text{shape} \mid \text{object}) \]
\[ = \max_h P(\text{appearance} \mid h, \text{object}) p(\text{shape} \mid h, \text{object}) p(h \mid \text{object}) \]

\( h \): assignment of features to parts
Probabilistic model

\[ P(\text{image} \mid \text{object}) = P(\text{appearance, shape} \mid \text{object}) \]
\[ = \max_h P(\text{appearance} \mid h, \text{object}) p(\text{shape} \mid h, \text{object}) p(h \mid \text{object}) \]
Probabilistic model

\[ P(\text{image} \mid \text{object}) = P(\text{appearance}, \text{shape} \mid \text{object}) \]
\[ = \max_h P(\text{appearance} \mid h, \text{object}) p(\text{shape} \mid h, \text{object}) p(h \mid \text{object}) \]

Distribution over joint part positions

2D image space
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
  - **Con:** too many doublet probabilities to estimate

- **Spatial pyramids**
  - **Pro:** simple extension of a bag of features, works very well
  - **Con:** no geometric invariance

- **Implicit shape models**
  - **Pro:** can localize object, maintain translation and possibly scale invariance
  - **Con:** need supervised training data (known object positions and possibly segmentation masks)

- **Generative part-based models**
  - **Pro:** very nice conceptually
  - **Con:** combinatorial hypothesis search problem