Object Recognition

• Intro
• Recognition of 3D objects
  • Recognition of object categories:
    • Bag of world models
    • Part based models
    • 3D object categorization

Computer Vision: Algorithms and Applications. R. Szeliski
Pages 696-709
Categorical vs Single Instance
Challenges: intra-class variation
Challenges:

Variability due to:

- View point
- Illumination
- Occlusions
- Etc..
Basic properties

• Representation
  – How to represent an object category; which classification scheme?

• Learning
  – How to learn the classifier, given training data

• Recognition
  – How the classifier is to be used on novel data
Part 1: Bag-of-words models

This segment is based on the tutorial “Recognizing and Learning Object Categories: Year 2007”, by Prof. A. Torralba, R. Fergus and F. Li
Related works

• Early “bag of words” models: mostly texture recognition

• Hierarchical Bayesian models for documents (pLSA, LDA, etc.)
  – Hoffman 1999; Blei, Ng & Jordan, 2004; Teh, Jordan, Beal & Blei, 2004

• Object categorization
  – Csurka, Bray, Dance & Fan, 2004; Sivic, Russell, Efros, Freeman & Zisserman, 2005; Sudderth, Torralba, Freeman & Willsky, 2005;

• Natural scene categorization
  – Vogel & Schiele, 2004; Fei-Fei & Perona, 2005; Bosch, Zisserman & Munoz, 2006
Object \rightarrow \text{Bag of ‘words’}
Analogy to documents

Of all the sensory impressions proceeding to the brain, the visual experiences are the dominant ones. Our perception of the world around us is based essentially on the messages that reach the brain from our eyes. For a long time it was thought that the retinal image was transmitted point by point to visual centers in the brain, as a movie screen upon which the image in the eye was projected. Through the discoveries of Hubel and Wiesel we now know that behind the origin of the visual perception in the brain there is a considerably more complicated course of events. By following the visual impulses along their path to the various cell layers of the optical cortex, Hubel and Wiesel have been able to demonstrate that the message about the image falling on the retina undergoes a step-wise analysis in a system of nerve cells stored in columns. In this system each cell has its specific function and is responsible for a specific detail in the pattern of the retinal image.

China is forecasting a trade surplus of $90bn (£51bn) to $100bn this year, a threefold increase on 2004’s $32bn. The Commerce Ministry said the surplus would be created by a predicted 30% jump in exports to $750bn, compared with a 18% increase in imports to $660bn. This is likely to annoy the US, which has long argued that China’s exports are unfairly helped by a deliberately undervalued yuan. The US has also needed to boost domestic demand so more goods stayed within the country. China has increased the value of the yuan against the dollar by 2.1% in July and permitted it to trade within a narrow band, but the US wants the yuan to be allowed to rise freely. However, Beijing has made it clear that it will take its time and tread carefully before allowing the yuan to rise further in value.
definition of “BoW”

– Independent features
definition of “BoW”

– Independent features
– histogram representation

codewords dictionary
1. Feature detection and representation
1. Feature detection and representation

- Regular grid
  - Vogel & Schiele, 2003
  - Fei-Fei & Perona, 2005
1. Feature detection and representation

- Regular grid
  - Vogel & Schiele, 2003
  - Fei-Fei & Perona, 2005
- Interest point detector
  - Csurka, et al. 2004
  - Fei-Fei & Perona, 2005
  - Sivic, et al. 2005
1. Feature detection and representation

- Regular grid
  - Vogel & Schiele, 2003
  - Fei-Fei & Perona, 2005
- Interest point detector
  - Csurka, Bray, Dance & Fan, 2004
  - Fei-Fei & Perona, 2005
  - Sivic, Russell, Efros, Freeman & Zisserman, 2005
- Other methods
  - Random sampling (Vidal-Naquet & Ullman, 2002)
  - Segmentation based patches (Barnard, Duygulu, Forsyth, de Freitas, Blei, Jordan, 2003)
1. Feature detection and representation

- Detect patches
  - [Mikojaczyk and Schmid '02]
  - [Mata, Chum, Urban & Pajdla, '02]
  - [Sivic & Zisserman, '03]

- Normalize patch

- Compute SIFT descriptor
  - [Lowe '99]

Slide credit: Josef Sivic
1. Feature detection and representation
2. Codewords dictionary formation
Example: color feature

- RGB values for different parts of the image:
  - Red (R): 0, 15, 3
  - Green (G): 200, 189, 12
  - Blue (B): 20, 2, 2
Example: color feature

\[ b \]

\[ g \]

\[ r \]

\[ \begin{cases} R=15 \\ G=189 \\ B=2 \end{cases} \]
2. Codewords dictionary formation

Cluster center = code word

Clustering/vector quantization
2. Codewords dictionary formation
Image patch examples of codewords

Sivic et al. 2005
Visual vocabularies: Issues

• How to choose vocabulary size?
  – Too small: visual words not representative of all patches
  – Too large: quantization artifacts, overfitting

• Computational efficiency
  – Vocabulary trees
    (Nister & Stewenius, 2006)
3. Bag of word representation

- Nearest neighbors assignment
- K-D tree search strategy

Codewords dictionary
3. Bag of word representation

Codewords dictionary

frequency

codewords
Representing textures

- Texture is characterized by the repetition of basic elements or *textons*.
- For stochastic textures, it is the identity of the textons, not their spatial arrangement, that matters.

---


Credit slide: S. Lazebnik
Representing textures


Credit slide: S. Lazebnik
1. feature detection & representation

2. codewords dictionary

3. category models

Representation
Invariance issues

- Scale – rotation – view point - occlusions
  - Implicit
  - Detectors and descriptors

Kadir and Brady. 2003
Category models
Learning and Recognition

codewords dictionary

category models (and/or) classifiers

category decision
Learning and Recognition

1. Discriminative method:
   - NN
   - SVM

2. Generative method:
   - graphical models
Discriminative classifiers

category models

Model space
Discriminative classifiers

Query image

Winning class: pink

Model space
Nearest Neighbors classifier

Query image

Model space

Winning class: pink

- Assign label of nearest training data point to each test data point
K-Nearest Neighbors classifier

Query image

Model space

Winning class: pink

- For a new point, find the k closest points from training data
- Labels of the k points “vote” to classify
- Works well provided there is lots of data and the distance function is good
• Voronoi partitioning of feature space for 2-category 2-D and 3-D data

• For k dimensions: k-D tree = space-partitioning data structure for organizing points in a $k$-dimensional space
• Enable efficient search
Functions for comparing histograms

- **L1 distance**
  \[
  D(h_1, h_2) = \sum_{i=1}^{N} |h_1(i) - h_2(i)|
  \]

- **\(\chi^2\) distance**
  \[
  D(h_1, h_2) = \sum_{i=1}^{N} \frac{(h_1(i) - h_2(i))^2}{h_1(i) + h_2(i)}
  \]

- **Quadratic distance (cross-bin)**
  \[
  D(h_1, h_2) = \sum_{i,j} A_{ij} (h_1(i) - h_2(j))^2
  \]

Learning and Recognition

1. Discriminative method:
   - NN
   - SVM

2. Generative method:
   - graphical models
Discriminative classifiers (linear classifier)

category models

Model space

Class 1

Class N
Linear classifiers

• Find linear function (*hyperplane*) to separate positive and negative examples

\[ x_i \text{ positive: } x_i \cdot w + b \geq 0 \]
\[ x_i \text{ negative: } x_i \cdot w + b < 0 \]

Which hyperplane is best?

\( w, b \)
Support vector machines

• Find hyperplane that maximizes the *margin* between the positive and negative examples

Support vectors:

\[ x_i \cdot w + b = \pm 1 \]

Distance between point and hyperplane:

\[ \frac{|x_i \cdot w + b|}{||w||} \]

Margin = \( 2 / ||w|| \)

Solution:

\[ w = \sum_i \alpha_i y_i x_i \]

Classification function (decision boundary):

\[ w \cdot x + b = \sum_i \alpha_i y_i x_i \cdot x + b \]

Credit slide: S. Lazebnik
Support vector machines

• Classification

\[ \mathbf{w} \cdot \mathbf{x} + b = \sum_i \alpha_i y_i \mathbf{x}_i \cdot \mathbf{x} + b \]

Test point

if \( \mathbf{x} \cdot \mathbf{w} + b \geq 0 \) \( \rightarrow \) class 1

if \( \mathbf{x} \cdot \mathbf{w} + b < 0 \) \( \rightarrow \) class 2
Nonlinear SVMs

- Datasets that are linearly separable work out great:

- But what if the dataset is just too hard?

- We can map it to a higher-dimensional space:
Nonlinear SVMs

- General idea: the original input space can always be mapped to some higher-dimensional feature space where the training set is separable:

\[ \Phi: x \rightarrow \varphi(x) \]
Nonlinear SVMs

• Nonlinear decision boundary in the original feature space:

\[
\sum_i \alpha_i y_i K(x_i, x) + b
\]

• The kernel \( K = \) product of the lifting transformation \( \phi(x) \):

\[
K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)
\]

NOTE:
• It is not required to compute \( \phi(x) \) explicitly:
• The kernel must satisfy the “Mercer inequality”

C. Burges, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, 1998
Kernels for bags of features

• Histogram intersection kernel:

\[ I(h_1, h_2) = \sum_{i=1}^{N} \min(h_1(i), h_2(i)) \]

• Generalized Gaussian kernel:

\[ K(h_1, h_2) = \exp \left( -\frac{1}{A} D(h_1, h_2)^2 \right) \]

• \( D \) can be Euclidean distance, \( \chi^2 \) distance etc…

What about multi-class SVMs?

- No “definitive” multi-class SVM formulation
- In practice, we have to obtain a multi-class SVM by combining multiple two-class SVMs
- One vs. others
  - Training: learn an SVM for each class vs. the others
  - Testing: apply each SVM to test example and assign to it the class of the SVM that returns the highest decision value
- One vs. one
  - Training: learn an SVM for each pair of classes
  - Testing: each learned SVM “votes” for a class to assign to the test example
SVMs: Pros and cons

• Pros
  – Many publicly available SVM packages: http://www.kernel-machines.org/software
  – Kernel-based framework is very powerful, flexible
  – SVMs work very well in practice, even with very small training sample sizes

• Cons
  – No “direct” multi-class SVM, must combine two-class SVMs
  – Computation, memory
    • During training time, must compute matrix of kernel values for every pair of examples
    • Learning can take a very long time for large-scale problems
Object recognition results

- ETH-80 database
  8 object classes
  \( (Eichhorn\ and\ Chapelle\ 2004) \)

- Features:
  - Harris detector
  - PCA-SIFT descriptor, \( d=10 \)

<table>
<thead>
<tr>
<th>Kernel</th>
<th>Complexity</th>
<th>Recognition rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Match [Wallraven et al.]</td>
<td>( O(dm^2) )</td>
<td>84%</td>
</tr>
<tr>
<td>Bhattacharyya affinity [Kondor &amp; Jebara]</td>
<td>( O(dm^3) )</td>
<td>85%</td>
</tr>
<tr>
<td>Pyramid match</td>
<td>( O(dmL) )</td>
<td>84%</td>
</tr>
</tbody>
</table>

Slide credit: Kristen Grauman
Pyramid match kernel

- Fast approximation of Earth Mover’s Distance
- Weighted sum of histogram intersections at multiple resolutions (linear in the number of features instead of cubic)

Spatial Pyramid Matching

Discriminative models

**Nearest neighbor**

Shakhnarovich, Viola, Darrell 2003
Berg, Berg, Malik 2005...

**Support Vector Machines**

Guyon, Vapnik, Heisele, Serre, Poggio...

**Latent SVM**

Felzenszwalb 00
Ramanan 03...

**Neural networks**

LeCun, Bottou, Bengio, Haffner 1998
Rowley, Baluja, Kanade 1998
...

**Boosting**

Viola, Jones 2001,
Torralba et al. 2004,
Opelt et al. 2006,...
Learning and Recognition

1. Discriminative method:
   - NN
   - SVM

2. Generative method:
   - graphical models

→ Model the probability distribution that produces a given bag of features
Generative models

1. Naïve Bayes classifier
   - Csurka Bray, Dance & Fan, 2004

2. Hierarchical Bayesian text models (pLSA and LDA)
   - Background: Hoffman 2001, Blei, Ng & Jordan, 2004
   - Natural scene categorization: Fei-Fei et al. 2005
Object categorization: the statistical viewpoint

- Discriminative methods model posterior

- Generative methods model likelihood and prior

- Bayes rule:

\[
\frac{p(\text{zebra} \mid \text{image})}{p(\text{no zebra} \mid \text{image})} = \frac{p(\text{image} \mid \text{zebra})}{p(\text{image} \mid \text{no zebra})} \cdot \frac{p(\text{zebra})}{p(\text{no zebra})}
\]

posterior ratio likelihood ratio prior ratio
Some notations

- **w**: a collection of all $N$ codewords in the image
  
  $w = [w_1, w_2, \ldots, w_N]$

- **c**: category of the image
the Naïve Bayes model

\[ p(c \mid w) \sim p(c)p(w \mid c) = p(c) \ p(w_1, \ldots, w_N \mid c) \]

Prior prob. of the object classes
Image likelihood given the class
the Naïve Bayes model

$$p(c \mid w) \sim p(c)p(w \mid c) = p(c) p(w_1, \ldots, w_N \mid c)$$

$$= p(c) \prod_{n=1}^{N} p(w_n \mid c)$$

- Assume that each feature (codewords) is conditionally independent given the class

$$p(w_1, \ldots, w_N \mid c) = \prod_{i=1}^{N} p(w_i \mid c)$$
the Naïve Bayes model

\[ p(c \mid w) \sim p(c)p(w \mid c) = p(c) \ p(w_1, \ldots, w_N \mid c) \]

Prior prob. of the object classes

Image likelihood given the class

Likelihood of \( n^{\text{th}} \) visual word given the class
Classification/Recognition

\[ c^* = \arg \max_c p(c \mid w) \propto p(c) p(w \mid c) = p(c) \prod_{n=1}^{N} p(w_n \mid c) \]

Object class decision

- How do we learn \( P(w_i \mid c_j) \)?
- From empirical frequencies of code words in images from a given class
Our in-house database contains 1776 images in seven classes: faces, buildings, trees, cars, phones, bikes and books. Fig. 2 shows some examples from this dataset.
Table 1. Confusion matrix and the mean rank for the best vocabulary ($k=1000$).

<table>
<thead>
<tr>
<th>True classes</th>
<th>faces</th>
<th>buildings</th>
<th>trees</th>
<th>cars</th>
<th>phones</th>
<th>bikes</th>
<th>books</th>
</tr>
</thead>
<tbody>
<tr>
<td>faces</td>
<td>76</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>13</td>
</tr>
<tr>
<td>buildings</td>
<td>2</td>
<td>44</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>trees</td>
<td>3</td>
<td>2</td>
<td>80</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>cars</td>
<td>4</td>
<td>1</td>
<td>0</td>
<td>75</td>
<td>3</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>phones</td>
<td>9</td>
<td>15</td>
<td>1</td>
<td>16</td>
<td>70</td>
<td>14</td>
<td>11</td>
</tr>
<tr>
<td>bikes</td>
<td>2</td>
<td>15</td>
<td>12</td>
<td>0</td>
<td>8</td>
<td>73</td>
<td>0</td>
</tr>
<tr>
<td>books</td>
<td>4</td>
<td>19</td>
<td>0</td>
<td>6</td>
<td>7</td>
<td>2</td>
<td>69</td>
</tr>
<tr>
<td>Mean ranks</td>
<td>1.49</td>
<td>1.88</td>
<td>1.33</td>
<td>1.33</td>
<td>1.63</td>
<td>1.57</td>
<td>1.57</td>
</tr>
</tbody>
</table>

$E = 28\%$

Table 2. Confusion matrix and mean rank for SVM ($k=1000$, linear kernel).

<table>
<thead>
<tr>
<th>True classes</th>
<th>faces</th>
<th>buildings</th>
<th>trees</th>
<th>cars</th>
<th>phones</th>
<th>bikes</th>
<th>books</th>
</tr>
</thead>
<tbody>
<tr>
<td>faces</td>
<td>98</td>
<td>14</td>
<td>10</td>
<td>10</td>
<td>34</td>
<td>0</td>
<td>13</td>
</tr>
<tr>
<td>buildings</td>
<td>1</td>
<td>63</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>trees</td>
<td>1</td>
<td>10</td>
<td>81</td>
<td>1</td>
<td>0</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>cars</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>85</td>
<td>5</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>phones</td>
<td>0</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>55</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>bikes</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>91</td>
<td>0</td>
</tr>
<tr>
<td>books</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>73</td>
</tr>
<tr>
<td>Mean ranks</td>
<td>1.04</td>
<td>1.77</td>
<td>1.28</td>
<td>1.30</td>
<td>1.83</td>
<td>1.09</td>
<td>1.39</td>
</tr>
</tbody>
</table>

$E = 15\%$
Summary: Generative models

- Naïve Bayes
  - *Unigram models* in document analysis
  - Assumes conditional independence of words given class
  - Parameter estimation: frequency counting
Other generative BoW models

• Hierarchical Bayesian topic models (e.g. pLSA and LDA)
  – Natural scene categorization: Fei-Fei et al. 2005
Generative vs discriminative

• Discriminative methods
  – Computationally efficient & fast

• Generative models
  – Convenient for weakly- or un-supervised, incremental training
  – Prior information
  – Flexibility in modeling parameters
Weakness of BoW the models

• No rigorous geometric information of the object components
• It’s intuitive to most of us that objects are made of parts – no such information
• Not extensively tested yet for
  – View point invariance
  – Scale invariance
• Segmentation and localization unclear
EECS 442 – Computer vision

Object Recognition

• Intro
• Recognition of 3D objects
• Recognition of object categories:
  • Bag of world models
  • Part based models
  • 3D object categorization