CS688/WST665: Web-Scale Image Retrieval
Bag-of-Words (BoW) Models

Sung-Eui Yoon
(윤성의)

Course URL:
http://sglab.kaist.ac.kr/~sungeui/IR
What we will learn today?

- Bag of Words models
  - Basic representation
  - Different learning and recognition algorithms
Object $\rightarrow$ 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 us through our eyes. For a long time, the retina image was the key to visual centers in the brain. However, as neuroscience advanced, a movie screen image could be discovered. We know that the perception of the retina is much more complicated. Following the path to the various centers of the cerebral cortex, Hubel and Wiesel have demonstrated that the message about the image falling on the retina undergoes a hierarchical analysis in a system of nerve cells. This is 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% increase in exports to $750bn, compared with $560bn in 2004. The US dollar is $660bn. The law would allow the yuan to be delisted by the US. China’s government also needs to satisfy demand so its currency will rise. China has already started to keep the yuan against the dollar and permitted it to trade within a narrow range, but the US wants the yuan to be allowed to trade freely. However, Beijing has made it clear that it will take its time and tread carefully by allowing the yuan to rise carefully.

Fei-Fei Li
definition of “BoW”

- Independent features

face

bike

violin

Fei-Fei Li
definition of “BoW”

- Independent features
- histogram representation
Representation

feature detection & representation

image representation

Codewords dictionary

category models (and/or) classifiers

category decision
1. Feature detection and representation
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

- Compute SIFT descriptor
  [Lowe '99]

- Normalize patch

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

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

- Cluster center = code word
- Clustering/vector quantization

Fei-Fei Li
K-Means Clustering

- Minimizing the within-cluster sum of squares (WCSS)

\[
\arg\min_s \sum_{i=1}^{k} \sum_{x_j \in S_i} \|x_j - \mu_i\|^2
\]

Demonstration of the standard algorithm

1) k initial "means" (in this case k=3) are randomly selected from the data set (shown in color).
2) k clusters are created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means.
3) The centroid of each of the k clusters becomes the new means.
4) Steps 2 and 3 are repeated until convergence has been reached.
2. Codewords dictionary formation
Image patch examples of codewords

Fei-Fei Li

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

[Diagram showing a bag of word representation with codewords and frequency bars, along with a codewords dictionary graph.]
Representation

1. feature detection & representation

2. codewords dictionary

3. image representation

Fei-Fei Li
Learning and Recognition

codewords dictionary

category models (and/or) classifiers

category decision
PA2

- Understand and implement a basic image retrieval system
- Use the original UKBenchmark

<table>
<thead>
<tr>
<th>Query</th>
<th>First</th>
<th>Second</th>
<th>Third</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="11.447705.jpg" alt="Query Image" /></td>
<td><img src="28.526271.jpg" alt="First Image" /></td>
<td><img src="29.273746.jpg" alt="Second Image" /></td>
<td><img src="31.938790.jpg" alt="Third Image" /></td>
</tr>
</tbody>
</table>

![Additional Images](11.447705.jpg)  ![Additional Images](28.526271.jpg)  ![Additional Images](29.273746.jpg)  ![Additional Images](31.938790.jpg)
Learning and Recognition

1. Discriminative method:
   - NN
   - SVM

2. Generative method:
   - graphical models

category models (and/or) classifiers
Discriminative classifiers

category models

Model space

Class 1

Class N

Fei-Fei Li
Discriminative classifiers

Query image

Winning class: pink

Model space
Nearest Neighbors classifier

Query image

Model space

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

Query image

Winning class: pink

Model space

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
K-Nearest Neighbors classifier

- 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

Fei-Fei Li
Overview of kd-Trees

- Binary spatial subdivision (special case of BSP tree)
- Split planes aligned on main axis
- Inner nodes: subdivision planes
- Leaf nodes: points
2D Example with Triangles
2D Example with Triangles
2D Example with Triangles
2D Example with Triangles
Split Planes

- How to select axis & split plane?

  - Option 1:
    - Choose a random dimension
    - Subdivide in the middle

  - Option 2:
    - Choose a dimension that has a high variance

  - Any other options
Nearest Neighbor Search with \textit{kd-tree}

- \textbf{Goal:} find \( k \) nearest neighbors given a point
  - Commonly identify approximate, not exact nearest neighbors

- Apply a depth-first search
  - Traverse the tree with a stack

- Or, we can apply a best-bin first search
  - Traverse more promising nodes first

- Traverse until we visit a certain number of nodes
Hashing techniques

- Kd-trees are not scalable
- Hashing arise as better technology
Functions for comparing histograms

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

- **\(\chi^2\) distance**
  (Variance of count data follows mean of the \(m\) usually)
  \[
  D(h_1, h_2) = \sum_{i=1}^{N} \frac{(h_1(i) - h_2(i))^2}{h_1(i) + h_2(i)}
  \]

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

---


Fei-Fei Li
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

Fei-Fei Li
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 = \( \frac{2}{||w||} \)

Solution: \( w = \sum_i \alpha_i y_i x_i \)

Classification function (decision boundary): \( w \cdot x + b = \sum_i \alpha_i y_i x_i \cdot x + b \)
where \( x_\perp \) is the normal projection of \( x \) onto the hyperplane and \( r \) gives us the distance from \( x \) to the hyperplane, negative if \( x \) is on the negative side, and positive if \( x \) is on the positive side (see figure 10.2). Calculating \( g(x) \) and noting that \( g(x_\perp) = 0 \), we have

\[
(10.4) \quad r = \frac{g(x)}{|w|}
\]

We see then that the distance to origin is

\[
(10.5) \quad r_0 = \frac{w_0}{|w|}
\]

Thus \( w_0 \) determines the location of the hyperplane with respect to the origin, and \( w \) determines its orientation.

10.3.2 Multiple Classes

When there are \( K > 2 \) classes, there are \( K \) discriminant functions. When they are linear, we have

\[
(10.6) \quad g_i(x) = w_i^T x + w_{i0}
\]

\[
g(x) = w_1^T x_1 + w_2^T x_2 + w_3^T x_3 + w_0.
\]
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

\[ \text{if } \mathbf{x} \cdot \mathbf{w} + b \geq 0 \rightarrow \text{ class 1} \]
\[ \text{if } \mathbf{x} \cdot \mathbf{w} + b < 0 \rightarrow \text{ class 2} \]


Fei-Fei Li
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:

Slide credit: Andrew Moore
Nonlinear SVMs

- General idea: the original input space can always be mapped to some higher-dimensional feature space where the training set is separable:
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 of 8 object classes  
  \textit{(Eichhorn and Chapelle 2004)}
- Features:
  - Harris detector
  - PCA-SIFT descriptor, $d=10$

- Achieves a high accuracy (about 80%)
Discriminative models

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

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

Support Vector Machines
Guyon, Vapnik, Heisele, Serre, Poggio...

Latent SVM
Structural SVM
Felzenszwalb 00
Ramanan 03...

Boosting
Viola, Jones 2001,
Torralba et al. 2004,
Opelt et al. 2006,
...

Source: Vittorio Ferrari, Kristen Grauman, Antonio Torralba

Fei-Fei Li
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

Fei-Fei Li
Some notations

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

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

\[ \text{Posterior} = p(c | w) \propto p(c)p(w | c) \]

- Probability that image \( I \) is of category \( c \)
- Prior probability of the object classes
- Image likelihood given the class
the Naïve Bayes model

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

Object class decision

Likelihood of ith visual word given the class

Estimated by 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><strong>faces</strong></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><strong>buildings</strong></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><strong>trees</strong></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><strong>cars</strong></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><strong>phones</strong></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><strong>bikes</strong></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><strong>books</strong></td>
<td>4</td>
<td>19</td>
<td>0</td>
<td>6</td>
<td>7</td>
<td>2</td>
<td>69</td>
</tr>
</tbody>
</table>

Csurka et al. 2004

Fei-Fei Li
Generative vs discriminative

• Discriminative methods
  – Computationally efficient & fast

• Generative models
  – 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
What have we learned today?

• Bag of Words models
  – Basic representation
  – Different learning and recognition algorithms
Next Time...

- Various image retrieval systems
Homework for Every Class

- Go over the next lecture slides
- Come up with one question on what we have discussed today
  - 1 for typical questions (that were answered in the class)
  - 2 for questions with thoughts or that surprised me
- Write questions at least 4 times