Bag-of-Visual-Words

16-385 Computer Vision
Carnegie Mellon University (Kris Kitani)
What object do these parts belong to?
Some local features are very informative.

An object as a collection of local features (bag-of-features):

- deals well with occlusion
- scale invariant
- rotation invariant
(not so) crazy assumption

spatial information of local features can be ignored for object recognition (i.e., verification)
Csurka et al. (2004), Willamowski et al. (2005), Grauman & Darrell (2005), Sivic et al. (2003, 2005)

Works pretty well for image-level classification

<table>
<thead>
<tr>
<th>class</th>
<th>bag of features</th>
<th>bag of features</th>
<th>Parts-and-shape model</th>
</tr>
</thead>
<tbody>
<tr>
<td>airplanes</td>
<td>98.8</td>
<td>97.1</td>
<td>90.2</td>
</tr>
<tr>
<td>cars (rear)</td>
<td>98.3</td>
<td>98.6</td>
<td>90.3</td>
</tr>
<tr>
<td>cars (side)</td>
<td>95.0</td>
<td>87.3</td>
<td>88.5</td>
</tr>
<tr>
<td>faces</td>
<td>100</td>
<td>99.3</td>
<td>96.4</td>
</tr>
<tr>
<td>motorbikes</td>
<td>98.5</td>
<td>98.0</td>
<td>92.5</td>
</tr>
<tr>
<td>spotted cats</td>
<td>97.0</td>
<td>—</td>
<td>90.0</td>
</tr>
</tbody>
</table>
Bag-of-features represent a data item (document, texture, image) as a histogram over features.
Bag-of-features

represent a data item (document, texture, image) as a histogram over features

an old idea
(e.g., texture recognition and information retrieval)
Texture recognition

Mori, Belongie and Malik, 2001

Julesz, 1981
Vector Space Model


http://www.todley.com/generatorsnewspaper/snippet.asp
A document (datapoint) is a vector of counts over each word (feature)

\[ \mathbf{v}_d = [n(w_{1,d}) \hspace{0.2cm} n(w_{2,d}) \hspace{0.2cm} \cdots \hspace{0.2cm} n(w_{T,d})] \]

\( n(\cdot) \) counts the number of occurrences

just a histogram over words

What is the similarity between two documents?
A document (datapoint) is a vector of counts over each word (feature)

\[ \mathbf{v}_d = [n(w_{1,d}) \ n(w_{2,d}) \ \cdots \ n(w_{T,d})] \]

\( n(\cdot) \) counts the number of occurrences

just a histogram over words

What is the similarity between two documents?

Use any distance you want but the cosine distance is fast.

\[ d(\mathbf{v}_i, \mathbf{v}_j) = \cos \theta \]

\[ = \frac{\mathbf{v}_i \cdot \mathbf{v}_j}{\|\mathbf{v}_i\| \|\mathbf{v}_j\|} \]
but not all words are created equal
TF-IDF

Term frequency Inverse Document Frequency

$$\mathbf{v}_d = [n(w_1,d) \ n(w_2,d) \ \cdots \ n(w_T,d)]$$

but not all words are created equal

$$\mathbf{v}_d = [n(w_1,d)\alpha_1 \ n(w_2,d)\alpha_2 \ \cdots \ n(w_T,d)\alpha_T]$$

$$n(w_i,d)\alpha_i = n(w_i,d) \log \left\{ \frac{D}{\sum_{d'} 1[w_i \in d']} \right\}$$

- term frequency
- inverse document frequency
Example of tf–idf

Suppose we have term frequency tables for a collection consisting of only two documents, as listed on the right, then calculation of tf–idf for the term "this" in document 1 is performed as follows.

Tf, in its basic form, is just the frequency that we look up in appropriate table. In this case, it's one.

Idf is a bit more involved:

\[
idf(this, D) = \log \frac{N}{|\{d \in D : t \in d\}|}
\]

The numerator of the fraction is the number of documents, which is two. The number of documents in which "this" appears is also two, giving

\[
idf(this, D) = \log \frac{2}{2} = 0
\]

So tf–idf is zero for this term, and with the basic definition this is true of any term that occurs in all documents.

A slightly more interesting example arises from the word "example", which occurs three times but in only one document. For this document, tf–idf of "example" is:

\[
\begin{align*}
tf(example, d_2) &= 3 \\
idf(example, D) &= \log \frac{2}{1} \approx 0.6931 \\
tf(idf(example, d_2) &= tf(example, d_2) \times idf(example, D) = 3 \log 2 \approx 2.0794
\end{align*}
\]

(using the natural logarithm).
Standard BOW pipeline
(for image classification)
1. Extract features

2. Learn “visual vocabulary”

3. Quantize features using visual vocabulary

4. Represent images by frequencies of “visual words”
1. Extract features

2. Learn “visual vocabulary”

3. Quantize features using visual vocabulary

4. Represent images by frequencies of “visual words”
1. Extract features

2. Learn “visual vocabulary”

3. Quantize features using visual vocabulary

4. Represent images by frequencies of “visual words”
1. Extract features

2. Learn “visual vocabulary”

3. Quantize features using visual vocabulary

4. Represent images by frequencies of “visual words”
1. Extract features

2. Learn “visual vocabulary”

3. Quantize features using visual vocabulary

4. Represent images by frequencies of “visual words”
Feature Extraction
• Regular grid
  • Vogel & Schiele, 2003
  • Fei-Fei & Perona, 2005

• Interest point detector
  • Csurka et al. 2004
  • Fei-Fei & Perona, 2005
  • Sivic et al. 2005

• Other methods
  • Random sampling (Vidal-Naquet & Ullman, 2002)
  • Segmentation-based patches (Barnard et al. 2003)
Compute SIFT descriptor
[Lowe'99]

Normalize patch

Detect patches
[Mikojaczyk and Schmid '02]
[Mata, Chum, Urban & Pajdla, '02]
[Sivic & Zisserman, '03]
Visual Vocabulary
Clustering...
K-means Clustering

Given k:

1. Select initial centroids at random.

2. Assign each object to the cluster with the nearest centroid.

3. Compute each centroid as the mean of the objects assigned to it.

4. Repeat previous 2 steps until no change.
Clustering and vector quantization

- Clustering is a common method for learning a visual vocabulary or codebook
  - Unsupervised learning process
  - Each cluster center produced by k-means becomes a codevector
  - Codebook can be learned on separate training set
  - Provided the training set is sufficiently representative, the codebook will be “universal”

- The codebook is used for quantizing features
  - A *vector quantizer* takes a feature vector and maps it to the index of the nearest codevector in a codebook
  - Codebook = visual vocabulary
  - Codevector = visual word
Example visual vocabulary

Fei-Fei et al. 2005
Example codebook

Source: B. Leibe
Another codebook

Source: B. Leibe
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. Image representation

frequency

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>

codewords
Image classification

- Given the bag-of-features representations of images from different classes, learn a classifier using machine learning.
Extension to bag-of-words models
But what about layout?

All of these images have the same color histogram
Spatial pyramid representation

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

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

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