Bag of Words
or
Bag of Features
Lecture-16
(Slides Credit: Cordelia Schmid
LEAR – INRIA Grenoble)
Contents

• Interest Point Detector
• Interest Point Descriptor
• K-means clustering
• Support Vector Machine (SVM) classifier
• Evaluation Metrics: Precision & Recall
Image classification

- Image classification: assigning a class label to the image

Car: present
Cow: present
Bike: not present
Horse: not present
...

...
Image classification

- Image classification: assigning a class label to the image
  - Car: present
  - Cow: present
  - Bike: not present
  - Horse: not present
  - ...

- Object localization: define the location and the category
  - Location
  - Category
Difficulties: within object variations

Variability: Camera position, Illumination, Internal parameters

Within-object variations
Difficulties: within class variations
Image classification

- Given
  Positive training images containing an object class
  Negative training images that don’t

- Classify
  A test image as to whether it contains the object class or not
Bag-of-features – Origin: texture recognition

- Texture is characterized by the repetition of basic elements or *textons*

Bag-of-features – Origin: texture recognition

universal texton dictionary

histogram

The last duel
After quarrelling over a bank loan, two men took part in the last fatal duel staged on Scottish soil. BBC News's James Landale retraces the steps of his ancestor who made that final challenge.

West Bank water row
Palestinians have accused Israel of diverting water away from their towns in order to keep Jewish settlements in the occupied territories fully supplied. Israel denies the charge saying Palestinian farmers are to blame for using illegal connections to irrigate their fields.
Bag of Visual Words model

Image Dataset → Feature Detection → Local Patches
Bag of Visual Words model

Local Patches

Descriptors

Clustering

Generate the visual vocabulary

Words
Bag of Visual Words model

Represent an image as a histogram or bag of words
Bag-of-features – Origin: bag-of-words (text)

- Orderless document representation: frequencies of words from a dictionary
- Classification to determine document categories

---

Bag-of-words

<table>
<thead>
<tr>
<th></th>
<th>d1</th>
<th>d2</th>
<th>d3</th>
<th>d4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>People</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Sculpture</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

...
Bag-of-features for image classification

Extract regions or Interest Points

Compute descriptors

Find clusters and frequencies

Image 1

\[ S_1 \]
Bag-of-features for image classification

Step 1
- Extract regions
- Compute descriptors

Step 2
- Find clusters and frequencies
- Compute distance matrix

Step 3
- Classification

SVM
Step 1: feature extraction

- Detect Interest Points
  - SIFT
  - Harris
  - Dense (take every nth pixel as interest point)

- Compute Descriptor around each interest point
  - SIFT
  - HOG
Dense features
Bag-of-features for image classification

Step 1
- Extract regions
- Compute descriptors

Step 2
- Find clusters and frequencies

Step 3
- Compute distance matrix
- Classification

SVM
Step 2: Quantization
Step 2: Quantization

- Cluster descriptors
  - K-means

- Assign each visual word to a cluster

- Build frequency histogram
Choose $k$ data points to act as cluster centers

Until the cluster centers are unchanged

    Allocate each data point to cluster whose center is nearest

Replace the cluster centers with the mean of the elements in their clusters.

end

Algorithm 16.5: *Clustering by K-Means*
K-means Clustering: Step 1

Algorithm: k-means, Distance Metric: Euclidean Distance

From unknown source on internet
K-means Clustering: Step 2

Algorithm: k-means, Distance Metric: Euclidean Distance
K-means Clustering: Step 3
Algorithm: k-means, Distance Metric: Euclidean Distance

From unknown source on internet
K-means Clustering: Step 4

Algorithm: k-means, Distance Metric: Euclidean Distance
K-means Clustering: Step 5

Algorithm: k-means, Distance Metric: Euclidean Distance

From unknown source on internet
Example: 3-means Clustering

Convergence in 3 steps

from Duda et al.
Examples for visual words

<table>
<thead>
<tr>
<th>Category</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airplanes</td>
<td><img src="image" alt="Airplanes Examples" /></td>
</tr>
<tr>
<td>Motorbikes</td>
<td><img src="image" alt="Motorbikes Examples" /></td>
</tr>
<tr>
<td>Faces</td>
<td><img src="image" alt="Faces Examples" /></td>
</tr>
<tr>
<td>Wild Cats</td>
<td><img src="image" alt="Wild Cats Examples" /></td>
</tr>
<tr>
<td>Leaves</td>
<td><img src="image" alt="Leaves Examples" /></td>
</tr>
<tr>
<td>People</td>
<td><img src="image" alt="People Examples" /></td>
</tr>
<tr>
<td>Bikes</td>
<td><img src="image" alt="Bikes Examples" /></td>
</tr>
</tbody>
</table>
Image representation

Each image is represented by a vector, typically 1000-4000 dimension.
Bag-of-features for image classification

Step 1: Extract regions

Step 2: Compute descriptors and find clusters and frequencies

Step 3: Compute distance matrix

SVM: Classification
Step 3: Classification

- Learn a decision rule (classifier) assigning bag-of-features representations of images to different classes
Training data

Vectors are histograms, one from each training image

positive

Train classifier, e.g., SVM
Support Vector Machines (SVM)
Application

- Pattern recognition

- Object classification/detection
Usage

- The classifier must be trained using a set of negative and positive examples.

- The classifier “learns” the regularities in the data.

- If training was successful, the classifier is capable of classifying an unknown example with a high degree of accuracy.
Linear Classifier

- Binary classifier → Task of separating classes in feature space

\[
\begin{align*}
\mathbf{w}^T \mathbf{x} + b &= 0 \\
\mathbf{w}^T \mathbf{x} + b > 0 \\
\mathbf{w}^T \mathbf{x} + b < 0
\end{align*}
\]

\[
f(x) = \text{sign}(\mathbf{w}^T \mathbf{x} + b)
\]
Linear Classifier cont’d

Which of the linear separators is optimal?
Margin

- Distance from example to the separator is (Point to Plane Distance Equation)
  \[ r = \frac{w^T x + b}{\|w\|} \]

- Examples closest to the hyperplane are **support vectors**.

- **Margin** $2\gamma$ of the separator is the width of separation between classes.

\[ 2\gamma \]

\[ r \]
Maximum Margin Classification

- Maximizing the margin is good according to intuition.
- Implies that only support vectors are important; other training examples are ignorable.
LibSVM

SVM implementation

- http://www.csie.ntu.edu.tw/~cjlin/libsvm/
- http://www.cs.wisc.edu/dmi/svm/
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 \( \rightarrow \) RBF kernel

- \( D \) can be \( \chi^2 \) distance

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

- Earth mover’s distance
Combining features

• SVM with multi-channel chi-square kernel

\[ K(H_i, H_j) = \exp \left( - \sum_{c \in \mathcal{C}} \frac{1}{A_c} D_c(H_i, H_j) \right) \]

- Channel \( c \) is a combination of detector, descriptor
- \( D_c(H_i, H_j) \) is the chi-square distance between histograms
  \[ D_c(H_1, H_2) = \frac{1}{2} \sum_{i=1}^{m} \left[ (h_{1i} - h_{2i})^2 / (h_{1i} + h_{2i}) \right] \]
  - \( A_c \) is the mean value of the distances between all training sample
- Extension: learning of the weights, for example with Multiple Kernel Learning (MKL)

Multi-class SVMs

- Various direct formulations exist, but they are not widely used in practice. It is more common to obtain multi-class SVMs by combining two-class SVMs in various ways.

- One versus all:
  - Training: learn an SVM for each class versus 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 versus one:
  - Training: learn an SVM for each pair of classes
  - Testing: each learned SVM “votes” for a class to assign to the test example
Why does SVM learning work?

- Learns foreground and background visual words
  - foreground words – high weight
  - background words – low weight
Localization according to visual word probability

Illustration

- foreground word more probable
- background word more probable
Illustration

A linear SVM trained from positive and negative window descriptors

A few of the highest weighted descriptor vector dimensions (= 'PAS + tile')

+ lie on object boundary (= local shape structures common to many training exemplars)
Bag-of-features for image classification

- Excellent results in the presence of background clutter
Examples for misclassified images

Books - misclassified into faces, faces, buildings

Buildings - misclassified into faces, trees, trees

Cars - misclassified into buildings, phones, phones
Bag of visual words summary

• **Advantages:**
  – largely unaffected by position and orientation of object in image
  – fixed length vector irrespective of number of detections
  – very successful in classifying images according to the objects they contain

• **Disadvantages:**
  – no explicit use of configuration of visual word positions
  – poor at localizing objects within an image
Evaluation of image classification

- PASCAL VOC [05-10] datasets

- PASCAL VOC 2007
  - Training *and* test dataset available
  - Used to report state-of-the-art results
  - Collected January 2007 from Flickr
  - 500,000 images downloaded and random subset selected
  - 20 classes
  - Class labels per image + bounding boxes
  - 5011 training images, 4952 test images

- Evaluation measure: average precision
PASCAL 2007 dataset
PASCAL 2007 dataset
Evaluation Metrics

Ground Truth (GT)
Results of Method (RM)
True Positives (TP)
True Negatives (TN)
False Positives (FP)
False Negatives (FN)

precision = \frac{\text{GT} \cap \text{RM}}{\text{RM}} = \frac{\text{TP}}{\text{RM}}

precision = \frac{\text{GT} \cap \text{RM}}{\text{RM}} = \frac{\text{TP}}{\text{FP} + \text{TP}}

recall = \frac{\text{GT} \cap \text{RM}}{\text{GT}} = \frac{\text{TP}}{\text{GT}}

recall = \frac{\text{GT} \cap \text{RM}}{\text{GT}} = \frac{\text{TP}}{\text{TP} + \text{FN}}
Evaluation

- **Average Precision [TREC]** averages precision over the entire range of recall
  - Curve interpolated to reduce influence of “outliers”

![Graph showing precision-recall curve]

- A good score requires both high recall and high precision
- Application-independent
- Penalizes methods giving high precision but low recall
Results for PASCAL 2007

• Winner of PASCAL 2007 [Marszalek et al.] : mAP 59.4
  – Combination of several different channels (dense + interest points, SIFT + color descriptors, spatial grids)
  – Non-linear SVM with Gaussian kernel

• Multiple kernel learning [Yang et al. 2009] : mAP 62.2
  – Combination of several features
  – Group-based MKL approach

• Combining object localization and classification [Harzallah et al.’09] : mAP 63.5
  – Use detection results to improve classification

• .....
Spatial pyramid matching

- Add spatial information to the bag-of-features
- Perform matching in 2D image space

[Lazebnik, Schmid & Ponce, CVPR 2006]
Related work

Similar approaches:

- Subblock description [Szummer & Picard, 1997]
- SIFT [Lowe, 1999]
- GIST [Torralba et al., 2003]
Spatial pyramid representation

Locally orderless representation at several levels of spatial resolution
Spatial pyramid representation

Locally orderless representation at several levels of spatial resolution
Spatial pyramid representation

Locally orderless representation at several levels of spatial resolution
Spatial pyramid matching

- Combination of spatial levels with pyramid match kernel [Grauman & Darrell’05]
- Intersect histograms, more weight to finer grids
Scene dataset [Labzenik et al.’06]

4385 images
15 categories
Scene classification

<table>
<thead>
<tr>
<th>L</th>
<th>Single-level</th>
<th>Pyramid</th>
</tr>
</thead>
<tbody>
<tr>
<td>0(1x1)</td>
<td>72.2±0.6</td>
<td></td>
</tr>
<tr>
<td>1(2x2)</td>
<td>77.9±0.6</td>
<td>79.0 ±0.5</td>
</tr>
<tr>
<td>2(4x4)</td>
<td>79.4±0.3</td>
<td>81.1 ±0.3</td>
</tr>
<tr>
<td>3(8x8)</td>
<td>77.2±0.4</td>
<td>80.7 ±0.3</td>
</tr>
</tbody>
</table>
## Retrieval examples

<table>
<thead>
<tr>
<th>(a) kitchen</th>
<th>living room</th>
<th>living room</th>
<th>living room</th>
<th>office</th>
<th>living room</th>
<th>living room</th>
<th>living room</th>
<th>living room</th>
</tr>
</thead>
<tbody>
<tr>
<td>(b) kitchen</td>
<td></td>
<td></td>
<td></td>
<td>office</td>
<td></td>
<td></td>
<td></td>
<td>inside city</td>
</tr>
<tr>
<td>(c) store</td>
<td></td>
<td></td>
<td></td>
<td>mountain</td>
<td></td>
<td></td>
<td></td>
<td>forest</td>
</tr>
<tr>
<td>(d) tall bldg</td>
<td></td>
<td></td>
<td></td>
<td>inside city</td>
<td></td>
<td></td>
<td></td>
<td>inside city</td>
</tr>
<tr>
<td>(e) tall bldg</td>
<td>inside city</td>
<td></td>
<td></td>
<td>mountain</td>
<td></td>
<td></td>
<td></td>
<td>mountain</td>
</tr>
<tr>
<td>(f) inside city</td>
<td>inside city</td>
<td></td>
<td></td>
<td>tall bldg</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Category classification – CalTech101

<table>
<thead>
<tr>
<th>L</th>
<th>Single-level</th>
<th>Pyramid</th>
</tr>
</thead>
<tbody>
<tr>
<td>0(1x1)</td>
<td>41.2±1.2</td>
<td></td>
</tr>
<tr>
<td>1(2x2)</td>
<td>55.9±0.9</td>
<td>57.0 ±0.8</td>
</tr>
<tr>
<td>2(4x4)</td>
<td>63.6±0.9</td>
<td>64.6 ±0.8</td>
</tr>
<tr>
<td>3(8x8)</td>
<td>60.3±0.9</td>
<td>64.6 ±0.7</td>
</tr>
</tbody>
</table>
Discussion

• Summary
  – Spatial pyramid representation: appearance of local image patches + coarse global position information
  – Substantial improvement over bag of features
  – Depends on the similarity of image layout

• Extensions
  – Flexible, object-centered grid
Human Action Recognition
Weizmann Action Dataset

- 10 actions
- 9 actors per action
KTH Data Set

- Six Categories, 25 actors, 4 instances, 600 clips
UCF Sports Action Dataset

9 actions, 142 videos.
IXMAS Multi-view Data Set

- 13 action categories, 4 camera views, 10 actors, 3 instances.
UCF YouTube Action Dataset (UCF-11)
UCF50
Bag of Visual Words model (II)
Bag of Visual Words model (II)

- Feature Detector
- 3D cuboids extraction
- Feature Clustering
- e.g. 3D Harris, 3D SIFT
- Video-words histogram
- Visual vocabulary
- e.g. HOF, 3DHOG
Histogram of Optical Flow (HOF)
Optical flow provides very useful motion information for Action Recognition
Histogram of Optical flow (HOF)

- Very similar to HOG

- Optical flow has two components
  - $u$

- Optical flow Speed

  $$M = \sqrt{u^2 + v^2}$$

- Optical flow Direction

  $$\theta = \arctan\left(\frac{v}{u}\right)$$
Histogram of Optical flow (HOF)

- Quantize the orientation into 9 bins (0-180) without Interpolation
- The vote is magnitude
HOF Steps

- Compute optical flow over the whole video
- Detect interest point locations or use densely sampled locations
HOF Steps

- Extract space time volumes in the neighborhood of interest points.
  - Size of neighborhood can be: $32 \times 32 \times 15$ or $18 \times 18 \times 18$, etc.

- Further divide neighborhood into small cuboids.

- Compute HOF in each small cuboids

- Concatenate histograms of all cuboids to make HOF descriptor of an interest point.
HOF Steps

Video → Interest Point → Neighborhood → Cuboids

Make 1D HOF vector