note: black & white
Recap: Bag of Words for Large Scale Retrieval
Summary – large scale retrieval

• We want to do feature matching (project 2) with a billion images

• Problem: the all-pairs local feature matching is slow!
Visual words

- Example: each group of patches belongs to the same visual word

Figure from Sivic & Zisserman, ICCV 2003

Kristen Grauman
Summary – large scale retrieval

• We want to do feature matching (project 2) with a billion images

• Problem: the all-pairs local feature matching is slow!

• Problem: Finding the overlap in visual words based on the Bags of Features is still too slow!
  – Solution: inverted file index, one lookup per word.
**Inverted file index**

- New query image is mapped to indices of database images that share a word.

<table>
<thead>
<tr>
<th>Word #</th>
<th>Image #</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1, 2</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>91</td>
<td>2</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

Kristen Grauman
Summary – large scale retrieval

• We want to do feature matching (project 2) with a billion images

• Problem: the all-pairs local feature matching is slow!
  – Solution: quantize features and build bag of feature representation. Lossy! But spatial verification can help.

• Problem: Finding the overlap in visual words based on the Bags of Features is still too slow!
  – Solution: inverted file index, one lookup per word.

• Problem: Even quantizing the local features into a visual word is too slow!
  – Solution: vocabulary tree. Lossy!
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% rise 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. Beijing agrees the surplus is too high, but says the yuan is only one factor. Bank of China governor Zhou Xiaochuan said the country also needed to do more to boost domestic demand so that more goods stayed within the country. China 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 trade 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.
**tf-idf weighting**

- **Term frequency – inverse document frequency**
- Describe frame by frequency of each word within it, downweight words that appear often in the database
- (Standard weighting for text retrieval)

\[
t_i = \frac{n_{id}}{n_d} \log \left( \frac{N}{n_i} \right)
\]

- Number of occurrences of word \(i\) in document \(d\)
- Number of words in document \(d\)
- Total number of documents in database
- Number of documents word \(i\) occurs in, in whole database

Kristen Grauman
Query expansion

Query: *golf green*

Results:

- How can the grass on the *greens* at a *golf* course be so perfect?
- For example, a skilled *golfer* expects to reach the *green* on a par-four hole in ...
- Manufactures and sells synthetic *golf* putting *greens* and mats.

Irrelevant result can cause a `topic drift`:

Query Expansion

Results

Spatial verification

New query

Chum, Philbin, Sivic, Isard, Zisserman: Total Recall..., ICCV 2007

Slide credit: Ondrej Chum
Scoring retrieval quality

Query

Database size: 10 images
Relevant (total): 5 images

Results (ordered):

precision = #relevant / #returned
recall = #relevant / #total relevant

Slide credit: Ondrej Chum
• Sliding window detector must evaluate tens of thousands of location/scale combinations
• Faces are rare: 0–10 per image
• For computational efficiency, we should try to spend as little time as possible on the non-face windows
• A megapixel image has ~10^6 pixels and a comparable number of candidate face locations
• To avoid having a false positive in every image image, our false positive rate has to be less than 10^-6

James Hays

Many Slides from Lana Lazebnik
Face detection and recognition

Detection

Recognition

“Sally”
Consumer application: Apple iPhoto

http://www.apple.com/ilife/iphoto/
Consumer application: Apple iPhoto
Can be trained to recognize pets!

Consumer application: Apple iPhoto

Things iPhoto thinks are faces
Funny Nikon ads

"The Nikon S60 detects up to 12 faces."
Funny Nikon ads

"The Nikon S60 detects up to 12 faces."
Challenges of face detection

- Sliding window detector must evaluate tens of thousands of location/scale combinations
- Faces are rare: 0–10 per image
  - For computational efficiency, we should try to spend as little time as possible on the non-face windows
  - A megapixel image has $\sim 10^6$ pixels and a comparable number of candidate face locations
  - To avoid having a false positive in every image, our false positive rate has to be less than $10^{-6}$
The Viola/Jones Face Detector

- A seminal approach to real-time object detection
- Training is slow, but detection is very fast
- Key ideas
  - Integral images for fast feature evaluation
  - Boosting for feature selection
  - Attentional cascade for fast rejection of non-face windows


~8000 citations!
Image Features

“Rectangle filters”

\[ \text{Value} = \sum (\text{pixels in white area}) - \sum (\text{pixels in black area}) \]
Example
Fast computation with integral images

- The *integral image* computes a value at each pixel \((x,y)\) that is the sum of the pixel values above and to the left of \((x,y)\), inclusive.

- This can quickly be computed in one pass through the image.
Computing the integral image
Computing the integral image

Cumulative row sum: $s(x, y) = s(x-1, y) + i(x, y)$

Integral image: $ii(x, y) = ii(x, y-1) + s(x, y)$

MATLAB: $ii = 	ext{cumsum}(	ext{cumsum}(	ext{double}(i)), 2)$;
Computing sum within a rectangle

- Let $A, B, C, D$ be the values of the integral image at the corners of a rectangle.
- Then the sum of original image values within the rectangle can be computed as:
  \[ \text{sum} = A - B - C + D \]
- Only 3 additions are required for any size of rectangle!
Computing a rectangle feature
Feature selection

• For a 24x24 detection region, the number of possible rectangle features is \( \sim 160,000! \)
Feature selection

• For a 24x24 detection region, the number of possible rectangle features is ~160,000!
• At test time, it is impractical to evaluate the entire feature set
• Can we create a good classifier using just a small subset of all possible features?
• How to select such a subset?
Boosting

- *Boosting* is a classification scheme that combines *weak learners* into a more accurate *ensemble classifier*
- Weak learners based on rectangle filters:

\[
h_t(x) = \begin{cases} 
1 & \text{if } p_t f_t(x) > p_t \theta_t \\
0 & \text{otherwise}
\end{cases}
\]

- Ensemble classification function:

\[
C(x) = \begin{cases} 
1 & \text{if } \sum_{t=1}^{T} \alpha_t h_t(x) > \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\
0 & \text{otherwise}
\end{cases}
\]
Training procedure

• Initially, weight each training example equally

• In each boosting round:
  • Find the weak learner that achieves the lowest \textit{weighted} training error
  • Raise the weights of training examples misclassified by current weak learner

• Compute final classifier as linear combination of all weak learners (weight of each learner is directly proportional to its accuracy)
  • Exact formulas for re-weighting and combining weak learners depend on the particular boosting scheme (e.g., AdaBoost)

Boosting intuition

Weak Classifier 1

Slide credit: Paul Viola
Boosting illustration

Weights Increased
Boosting illustration

Weak Classifier 2
Boosting illustration

Weights Increased
Boosting illustration

Weak Classifier 3
Final classifier is a combination of weak classifiers
Boosting for face detection

- First two features selected by boosting:

  This feature combination can yield 100% detection rate and 50% false positive rate.
Boosting vs. SVM

• Advantages of boosting
  • Integrates classifier training with feature selection
  • Complexity of training is linear instead of quadratic in the number of training examples
  • Flexibility in the choice of weak learners, boosting scheme
  • Testing is fast
  • Easy to implement

• Disadvantages
  • Needs many training examples
  • Training is slow
  • Often doesn’t work as well as SVM (especially for many-class problems)
Boosting for face detection

- A 200-feature classifier can yield 95% detection rate and a false positive rate of 1 in 14084

![ROC curve for 200 feature classifier](image)

Not good enough!

Receiver operating characteristic (ROC) curve
Attentional cascade

• We start with simple classifiers which reject many of the negative sub-windows while detecting almost all positive sub-windows

• Positive response from the first classifier triggers the evaluation of a second (more complex) classifier, and so on

• A negative outcome at any point leads to the immediate rejection of the sub-window
Attentional cascade

- Chain classifiers that are progressively more complex and have lower false positive rates:

![Diagram of attentional cascade with three classifiers](image)

Receiver operating characteristic

- % False Pos
- % Detection

0 50 100

FACE

IMAGE SUB-WINDOW

Classifier 1

Classifier 2

Classifier 3

T T T

T F F

F F F

NON-FACE NON-FACE NON-FACE

FACE
Attentional cascade

- The detection rate and the false positive rate of the cascade are found by multiplying the respective rates of the individual stages.
- A detection rate of 0.9 and a false positive rate on the order of $10^{-6}$ can be achieved by a 10-stage cascade if each stage has a detection rate of 0.99 ($0.99^{10} \approx 0.9$) and a false positive rate of about 0.30 ($0.3^{10} \approx 6 \times 10^{-6}$).
Training the cascade

• Set target detection and false positive rates for each stage
• Keep adding features to the current stage until its target rates have been met
  • Need to lower AdaBoost threshold to maximize detection (as opposed to minimizing total classification error)
  • Test on a validation set
• If the overall false positive rate is not low enough, then add another stage
• Use false positives from current stage as the negative training examples for the next stage
The implemented system

- **Training Data**
  - 5000 faces
    - All frontal, rescaled to 24x24 pixels
  - 300 million non-faces
    - 9500 non-face images
  - Faces are normalized
    - Scale, translation
- **Many variations**
  - Across individuals
  - Illumination
  - Pose
System performance

• Training time: “weeks” on 466 MHz Sun workstation
• 38 layers, total of 6061 features
• Average of 10 features evaluated per window on test set
• “On a 700 Mhz Pentium III processor, the face detector can process a 384 by 288 pixel image in about .067 seconds”
  • 15 Hz
  • 15 times faster than previous detector of comparable accuracy (Rowley et al., 1998)
Output of Face Detector on Test Images
Other detection tasks

Facial Feature Localization

Profile Detection

Male vs. female
Profile Detection
Profile Features
Summary: Viola/Jones detector

- Rectangle features
- Integral images for fast computation
- Boosting for feature selection
- Attentional cascade for fast rejection of negative windows
Face Recognition

Face Recognition

Attributes for training

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Positive Examples</th>
<th>Negative Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian</td>
<td><img src="image1" alt="Positive Examples" /></td>
<td><img src="image2" alt="Negative Examples" /></td>
</tr>
<tr>
<td>Blond Hair</td>
<td><img src="image3" alt="Positive Examples" /></td>
<td><img src="image4" alt="Negative Examples" /></td>
</tr>
<tr>
<td>Child</td>
<td><img src="image5" alt="Positive Examples" /></td>
<td><img src="image6" alt="Negative Examples" /></td>
</tr>
<tr>
<td>Male</td>
<td><img src="image7" alt="Positive Examples" /></td>
<td><img src="image8" alt="Negative Examples" /></td>
</tr>
</tbody>
</table>

Similes for training

<table>
<thead>
<tr>
<th>Simile</th>
<th>Positive Examples</th>
<th>Negative Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>R1 Eyebrows</td>
<td><img src="image9" alt="Positive Examples" /></td>
<td><img src="image10" alt="Negative Examples" /></td>
</tr>
<tr>
<td>R1 Eyes</td>
<td><img src="image11" alt="Positive Examples" /></td>
<td><img src="image12" alt="Negative Examples" /></td>
</tr>
<tr>
<td>R1 Nose</td>
<td><img src="image13" alt="Positive Examples" /></td>
<td><img src="image14" alt="Negative Examples" /></td>
</tr>
<tr>
<td>R1 Mouth</td>
<td><img src="image15" alt="Positive Examples" /></td>
<td><img src="image16" alt="Negative Examples" /></td>
</tr>
<tr>
<td>R2 Eyebrows</td>
<td><img src="image17" alt="Positive Examples" /></td>
<td><img src="image18" alt="Negative Examples" /></td>
</tr>
<tr>
<td>R2 Eyes</td>
<td><img src="image19" alt="Positive Examples" /></td>
<td><img src="image20" alt="Negative Examples" /></td>
</tr>
<tr>
<td>R2 Nose</td>
<td><img src="image21" alt="Positive Examples" /></td>
<td><img src="image22" alt="Negative Examples" /></td>
</tr>
<tr>
<td>R2 Mouth</td>
<td><img src="image23" alt="Positive Examples" /></td>
<td><img src="image24" alt="Negative Examples" /></td>
</tr>
</tbody>
</table>