Transfer learning with convolutional networks
Challenge winner's accuracy

- 2010: 7-layer Convolutional Networks
- 2011: 19 layers
- 2012: 152 layers
- 2014: 19 layers
Transfer learning with convolutional networks

• What do we do for a new image classification problem?

• Key idea:
  • *Freeze* parameters in feature extractor
  • *Retrain* classifier
## Transfer learning with convolutional networks

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Non-Convnet Method</th>
<th>Non-Convnet perf</th>
<th>Pretrained convnet + classifier</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caltech 101</td>
<td>MKL</td>
<td>84.3</td>
<td>87.7</td>
<td>+3.4</td>
</tr>
<tr>
<td>VOC 2007</td>
<td>SIFT+FK</td>
<td>61.7</td>
<td>79.7</td>
<td>+18</td>
</tr>
<tr>
<td>CUB 200</td>
<td>SIFT+FK</td>
<td>18.8</td>
<td>61.0</td>
<td>+42.2</td>
</tr>
<tr>
<td>Aircraft</td>
<td>SIFT+FK</td>
<td>61.0</td>
<td>45.0</td>
<td>-16</td>
</tr>
<tr>
<td>Cars</td>
<td>SIFT+FK</td>
<td>59.2</td>
<td>36.5</td>
<td>-22.7</td>
</tr>
</tbody>
</table>
Why transfer learning?

• Availability of training data

• Computational cost

• Ability to pre-compute feature vectors and use for multiple tasks

• Con: NO end-to-end learning
Finetuning
Finetuning

Initialize with pre-trained, then train with low learning rate.
## Finetuning

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Non-Convnet Method</th>
<th>Non-Convnet perf</th>
<th>Pretrained convnet + classifier</th>
<th>Finetuned convnet</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Caltech 101</td>
<td>MKL</td>
<td>84.3</td>
<td>87.7</td>
<td>88.4</td>
<td>+4.1</td>
</tr>
<tr>
<td>VOC 2007</td>
<td>SIFT+FK</td>
<td>61.7</td>
<td>79.7</td>
<td>82.4</td>
<td>+20.7</td>
</tr>
<tr>
<td>CUB 200</td>
<td>SIFT+FK</td>
<td>18.8</td>
<td>61.0</td>
<td>70.4</td>
<td>+51.6</td>
</tr>
<tr>
<td>Aircraft</td>
<td>SIFT+FK</td>
<td>61.0</td>
<td>45.0</td>
<td>74.1</td>
<td>+13.1</td>
</tr>
<tr>
<td>Cars</td>
<td>SIFT+FK</td>
<td>59.2</td>
<td>36.5</td>
<td>79.8</td>
<td>+20.6</td>
</tr>
</tbody>
</table>
Visualizing convolutional networks
Receptive field

- Which input pixels does a particular unit in a feature map depend on?

[Diagram showing convolution with a 3x3 filter]
Receptive field

5x5 receptive field

convolve with 3 x 3 filter

3x3 receptive field

convolve with 3 x 3 filter
Receptive field

convolve with 3 x 3 filter, subsample
Receptive field

7x7 receptive field: union of 9 3x3 fields with stride of 2

convolve with 3 x 3 filter, subsample by factor 2

3x3 receptive field

convolve with 3 x 3 filter
Visualizing convolutional networks

- Take images for which a given unit in a feature map scores high
- Identify the receptive field for each.

Visualizing convolutional networks II

• Block regions of the image and classify

Visualizing convolutional networks II

• Image pixels important for classification = pixels when blocked cause misclassification

Object detection
The Task
Datasets

- Face detection
- One category: face
- Frontal faces
- Fairly rigid, unoccluded

Pedestrians

- One category: pedestrians
- Slight pose variations and small distortions
- Partial occlusions

Histories of Oriented Gradients for Human Detection. N. Dalal and B. Triggs. CVPR 2005
PASCAL VOC

• 20 categories
• 10K images
• Large pose variations, heavy occlusions
• Generic scenes
• Cleaned up performance metric
Coco

- 80 diverse categories
- 100K images
- Heavy occlusions, many objects per image, large scale variations
Evaluation metric
Matching detections to ground truth

\[ \text{IoU}(A, B) = \frac{|A \cap B|}{|A \cup B|} \]
Matching detections to ground truth

- Match detection to most similar ground truth
  - highest IoU
- If IoU > 50%, mark as correct
- If multiple detections map to same ground truth, mark only one as correct
- **Precision** = #correct detections / total detections
- **Recall** = #ground truth with matched detections / total ground truth
Tradeoff between precision and recall

- ML usually gives scores or probabilities, so threshold
- Too low threshold $\rightarrow$ too many detections $\rightarrow$ low precision, high recall
- Too high threshold $\rightarrow$ too few detections $\rightarrow$ high precision, low recall
- Right tradeoff depends on application
  - Detecting cancer cells in tissue: need high recall
  - Detecting edible mushrooms in forest: need high precision
Average precision

Precision vs. Recall graph with a point at (1,1).
Average precision

Precision     Recall
  1            1
Average average precision

• AP marks detections with overlap > 50% as correct
• But may need better localization
• Average AP across multiple overlap thresholds
• Confusingly, still called average precision
• Introduced in COCO
Mean and category-wise AP

• Every category evaluated independently
• Typically report mean AP averaged over all categories
• Confusingly called “mean Average Precision”, or “mAP”
Why is detection hard(er)?

• Precise localization
Why is detection hard(er)?

- Much larger impact of pose
Why is detection hard(er)?

- Occlusion makes localization difficult
Why is detection hard(er)?

- Counting
Why is detection hard(er)?

- Small objects
Detection as classification

- Run through every possible box and classify
- How many boxes?
  - Every pair of pixels = 1 box
  
  \[
  \binom{N}{2} = O(N^2)
  \]

  - For 300 x 500 image, \( N = 150K \)
  - \( 2.25 \times 10^{10} \) boxes!
Idea 1: scanning window

• Fix size
  • Can take a few different sizes

• Fixed stride

• Convolution with a filter
  • Classic: compute HOG features over entire image
Dealing with scale
Dealing with scale
Idea 2: Object proposals

- Use segmentation to produce ~5K candidates

Selective Search for Object Recognition
J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, A. W. M. Smeulders
In International Journal of Computer Vision 2013.
Idea 2: Object proposals

What makes for effective detection proposals? J. Hosang, R. Benenson, P. Dollar, B. Schiele. In TPAMI
A rapid rise in performance

[Source: http://pascallin.eecs.soton.ac.uk/challenges/VOC/voc20(07,08,09,10,11,12)/results/index.html]

Slide credit: Ross Girshick
Complexity and the plateau

[Source: http://pascallin.ics.soton.ac.uk/challenges/VOC/voc2007-12/results/index.html]

Slide credit: Ross Girshick
SIFT, HOG, LBP, ...

\[\text{Regionlets (2013)}\]
\[\text{SegDPM (2013)}\]
\[\text{DPM}^+, \text{MKL}^+, \text{Selective Search, DPM}^+, \text{MKL}\]

\[\text{DPM, HOG+BOW}\]
\[\text{DPM, MKL}\]

PASCAL VOC challenge dataset

[Regionlets. Wang et al. ICCV’ 13]  [SegDPM. Fidler et al. CVPR’ 13]

Slide credit: Ross Girshick
**R-CNN:** Regions with CNN features

![Graph showing mAP (%) for R-CNN across different years of the PASCAL VOC challenge dataset.](image)

- PASCAL VOC challenge dataset
- Slide credit: Ross Girshick
R-CNN: Regions with CNN features

Input image
Extract region proposals (~2k / image)
Compute CNN features
Classify regions (linear SVM)

Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation

R. Girshick, J. Donahue, T. Darrell, J. Malik
IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014

Slide credit: Ross Girshick
R-CNN at test time: Step 2

Input image

Extract region proposals (~2k / image)

Compute CNN features

Slide credit: Ross Girshick
R-CNN at test time: Step 2

Input image

Extract region proposals (~2k / image)

Crop

Scale (anisotropic)

Compute CNN features

227 x 227

a. Crop

b. Scale (anisotropic)

Slide credit: Ross Girshick
R-CNN at test time: Step 2

1. Crop
2. Scale (anisotropic)
3. Forward propagate

Output: “fc7” features

Input image → Extract region proposals (~2k / image) → Compute CNN features

Slide credit: Ross Girshick
R-CNN at test time: Step 3

1. Input image
2. Extract region proposals (~2k / image)
3. Compute CNN features
4. Classify regions

Warped proposal
4096-dimensional fc7 feature vector
linear classifiers (SVM or softmax)

Person? 1.6
Horse? -0.3

Slide credit: Ross Girshick
Step 4: Object proposal refinement

Original proposal

Predicted object bounding box

Linear regression on CNN features

Bounding-box regression

Slide credit: Ross Girshick
Bounding-box regression

\[ \Delta h \times (h + h) \]

\[ \Delta w \times (w + w) \]

\[ (\Delta x \times (w + x), \Delta y \times (h + h)) \]

Slide credit: Ross Girshick
# R-CNN results on PASCAL

<table>
<thead>
<tr>
<th>Reference systems</th>
<th>VOC 2007</th>
<th>VOC 2010</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPM v5 (Girshick et al. 2011)</td>
<td>33.7%</td>
<td>29.6%</td>
</tr>
<tr>
<td>UVA sel. search (Uijlings et al. 2013)</td>
<td></td>
<td>35.1%</td>
</tr>
<tr>
<td>Regionlets (Wang et al. 2013)</td>
<td>41.7%</td>
<td>39.7%</td>
</tr>
<tr>
<td>SegDPM (Fidler et al. 2013)</td>
<td></td>
<td>40.4%</td>
</tr>
</tbody>
</table>

metric: mean average precision (higher is better)

Slide credit: Ross Girshick
R-CNN results on PASCAL

<table>
<thead>
<tr>
<th>Method</th>
<th>VOC 2007</th>
<th>VOC 2010</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPM v5 (Girshick et al. 2011)</td>
<td>33.7%</td>
<td>29.6%</td>
</tr>
<tr>
<td>UVA sel. search (Uijlings et al. 2013)</td>
<td></td>
<td>35.1%</td>
</tr>
<tr>
<td>Regionlets (Wang et al. 2013)</td>
<td>41.7%</td>
<td>39.7%</td>
</tr>
<tr>
<td>SegDPM (Fidler et al. 2013)</td>
<td></td>
<td>40.4%</td>
</tr>
<tr>
<td>R-CNN</td>
<td>54.2%</td>
<td>50.2%</td>
</tr>
<tr>
<td>R-CNN + bbox regression</td>
<td>58.5%</td>
<td>53.7%</td>
</tr>
</tbody>
</table>

Slide credit: Ross Girshick
Training R-CNN

• Train convolutional network on ImageNet classification

• *Finetune* on detection
  • Classification problem!
  • Proposals with IoU > 50% are positives
  • Sample fixed proportion of positives in each batch because of imbalance
Other details - Non-max suppression

How do we deal with multiple detections on the same object?
Other details - Non-max suppression

- Go down the list of detections starting from highest scoring
- Eliminate any detection that overlaps highly with a higher scoring detection
- Separate, heuristic step