Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation

Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik

R-CNN: Regions with CNN features

What is R-CNN? R-CNN is a state-of-the-art object detection system that combines bottom-up region proposals with rich features learned by a convolutional neural network (CNN).


Feature learning with CNNs

SIFT, HOG, LBP, etc.: All previous PASCAL detection top-performers were based on these low-level features.

Feature learning: R-CNN is the first CNN-based feature learning method to achieve state-of-the-art object detection performance on PASCAL VOC.

A brief history of convolutional neural networks

Fukushima 1980 - Neocognitron
LeCun et al. 1988 - 1998 - Handwritten digit reading
Hinton, Lipton, Williams 1986 - "t" vs. "c" problem
Romero, Courville, Bengio 2011 - Convolutional networks
Visualizing and understanding CNNs -lecun2012a

CNNS for object detection

Wallach, Mnih, Marcu 2014 - Multi-scale face detection
Newell & Peng 1995 - Hand-drawing
LeCun, Huang, Bottou 2004 - No. 1 on MNIST
Ciresan et al. 2012 - Mnemonic detection
Serre et al. 2012 - Pedestrian detection
Seung et al. 2012 - Detection performance (PASCAL VOC 30.5%)

SuperVision CNN - Krizhevsky, Sutskever, Hinton 2012

- Breakthrough ImageNet classification results: 48.4% at 1-4K, 36.7% at 1-1K, 28.7% at 1-100

R-CNN at test time

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

Proposal refinement by regression

Linear regression on CNN features

Region proposal mechanisms
- Selective search (Uijlings et al. 2013) used in this work
- Many other choices: objectness (Alexe et al.), category independent object proposals (Endres & Hoiem), CPMC (Carreira & Simo-Serra), MCG (Abeel et al.), BING (Ming et al.)...

PASCAL VOC test results

<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>35.1%</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>40.4%</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>

ILSVRC13 detection

- R-CNN B6
- SegDPM v5
- UVA-Eviction
- NEC-MIL
- OverFeat
- Toronto
- SYSU_Vision
- GPU-UCLA
- Delta
- UIC-IF

ILSVRC2013 detection test set mAP

<table>
<thead>
<tr>
<th>Method</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN B6</td>
<td>31.4%</td>
</tr>
<tr>
<td>SegDPM v5</td>
<td>24.3%</td>
</tr>
<tr>
<td>UVA-Eviction</td>
<td>22.6%</td>
</tr>
<tr>
<td>NEC-MIL</td>
<td>19.9%</td>
</tr>
<tr>
<td>OverFeat</td>
<td>19.4%</td>
</tr>
<tr>
<td>Toronto</td>
<td>15.5%</td>
</tr>
<tr>
<td>SYSU_Vision</td>
<td>10.5%</td>
</tr>
<tr>
<td>GPU-UCLA</td>
<td>6.8%</td>
</tr>
<tr>
<td>Delta</td>
<td>6.1%</td>
</tr>
<tr>
<td>UIC-IF</td>
<td>5.8%</td>
</tr>
</tbody>
</table>

False positive analysis

<table>
<thead>
<tr>
<th>Method</th>
<th>False positive @ Bbox overlap</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-CNN</td>
<td>0.2%</td>
</tr>
<tr>
<td>SegDPM v5</td>
<td>0.1%</td>
</tr>
<tr>
<td>UVA-Eviction</td>
<td>0.05%</td>
</tr>
<tr>
<td>NEC-MIL</td>
<td>0.01%</td>
</tr>
<tr>
<td>OverFeat</td>
<td>0.02%</td>
</tr>
<tr>
<td>Toronto</td>
<td>0.12%</td>
</tr>
</tbody>
</table>

Training R-CNN

1. Pre-train CNN for image classification
2. Fine-tune CNN for PASCAL detection
3. Train per-class linear SVMs

R-CNN in practice

Get code & models!

Semantic segmentation

- R-CNN operates on region proposals
- Extension from detection to semantic segmentation is easy

Semantic segmentation results

- CPMC segments (Carreira & Simo-Serra)
- Compute CNN features on both "full" and "4x"

Feature visualizations

- Mixture of "grandmother cells" and distributed codes

Conclusions

- R-CNN dramatically outperforms all previous detection methods on PASCAL VOC
- R-CNN outperforms recent CNN-based detection methods, such as SegDPM, on ImageNet
- R-CNN is relatively simple and completely open source
- Detection speed is a manageable 10s / image
- Excellent scalability with number of classes
- Try R-CNN today! Get code & models at bit.ly/rcnn-cvpr14

A particularly pleasing detection (that was marked as an error)

Bicycle false positive #15

Aha! It’s an unannotated bicycle!

Look for our papers building on R-CNN

- Training R-CNN with weak supervision. Song et al. ICM’14
- Analysis of CNN training and features. Agrawal et al. ECCV’14
- 50.5% PASCAL segmentation accuracy. Harfane et al. ECCV’14
- R-CNN with parts for fine-grained recognition. Zhang et al. ECCV’14
- Detection and segmentation in RGB-D images. Gupta et al. ECCV’14