Advanced Machine Learning
Lecture 16
Convolutional Neural Networks II

14.01.2016

Bastian Leibe
RWTH Aachen
http://www.vision.rwth-aachen.de/
leibe@vision.rwth-aachen.de
Announcements

- Lecture evaluation
  - Please fill out the evaluation forms.
This Lecture: **Advanced Machine Learning**

- **Regression Approaches**
  - Linear Regression
  - Regularization (Ridge, Lasso)
  - Gaussian Processes

- **Learning with Latent Variables**
  - Prob. Distributions & Approx. Inference
  - Mixture Models
  - EM and Generalizations

- **Deep Learning**
  - Linear Discriminants
  - Neural Networks
  - Backpropagation & Optimization
  - CNNs, RNNs, RBMs, etc.
Topics of This Lecture

• Recap: CNNs

• CNN Architectures
  ➢ LeNet
  ➢ AlexNet
  ➢ VGGNet
  ➢ GoogLeNet

• Visualizing CNNs
  ➢ Visualizing CNN features
  ➢ Visualizing responses
  ➢ Visualizing learned structures

• Applications
Recap: Convolutional Neural Networks

- Neural network with specialized connectivity structure
  - Stack multiple stages of feature extractors
  - Higher stages compute more global, more invariant features
  - Classification layer at the end


Slide credit: Svetlana Lazebnik
Recap: Intuition of CNNs

• Convolutional net
  - Share the same parameters across different locations
  - Convolutions with learned kernels

• Learn *multiple* filters
  - E.g. 1000x1000 image
  - 100 filters
  - 10x10 filter size
  - ⇒ only 10k parameters

• Result: Response map
  - size: 1000x1000x100
  - Only memory, not params!
Recap: Convolution Layers

- All Neural Net activations arranged in 3 dimensions
  - Multiple neurons all looking at the same input region, stacked in depth
  - Form a single \([1 \times 1 \times \text{depth}]\) depth column in output volume.

Naming convention:
Recap: Activation Maps

Each activation map is a depth slice through the output volume.

5×5 filters

Activation maps
Recap: Pooling Layers

- **Effect:**
  - Make the representation smaller without losing too much information
  - Achieve robustness to translations

![Diagram of pooling layers](image)
Topics of This Lecture

- Recap: CNNs

- CNN Architectures
  - LeNet
  - AlexNet
  - VGGNet
  - GoogLeNet

- Visualizing CNNs
  - Visualizing CNN features
  - Visualizing responses
  - Visualizing learned structures

- Applications
Recap: ImageNet Challenge 2012

- **ImageNet**
  - ~14M labeled internet images
  - 20k classes
  - Human labels via Amazon Mechanical Turk

- **Challenge (ILSVRC)**
  - 1.2 million training images
  - 1000 classes
  - Goal: Predict ground-truth class within top-5 responses
  - Currently one of the top benchmarks in Computer Vision

[Deng et al., CVPR’09]
CNN Architectures: AlexNet (2012)

- Similar framework as LeNet, but
  - Bigger model (7 hidden layers, 650k units, 60M parameters)
  - More data ($10^6$ images instead of $10^3$)
  - GPU implementation
  - Better regularization and up-to-date tricks for training (Dropout)

ILSVRC 2012 Results

- AlexNet almost halved the error rate
  - 16.4% error (top-5) vs. 26.2% for the next best approach
  - A revolution in Computer Vision
  - Acquired by Google in Jan ‘13, deployed in Google+ in May ‘13
CNN Architectures: VGGNet (2015)

- **Main ideas**
  - Deeper network
  - Stacked convolutional layers with smaller filters (+ nonlinearity)
  - Detailed evaluation of all components

<table>
<thead>
<tr>
<th>ConvNet Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A</strong></td>
</tr>
<tr>
<td>11 weight layers</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>input (224 × 224 RGB image)</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv3-64</td>
</tr>
<tr>
<td>LRN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>maxpool</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv3-128</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>maxpool</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv3-256</td>
</tr>
<tr>
<td>conv3-256</td>
</tr>
<tr>
<td>conv1-256</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>maxpool</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv3-512</td>
</tr>
<tr>
<td>conv3-512</td>
</tr>
<tr>
<td>conv1-512</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>maxpool</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv3-512</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>maxpool</th>
</tr>
</thead>
<tbody>
<tr>
<td>FC-4096</td>
</tr>
</tbody>
</table>

| soft-max |
Comparison: AlexNet vs. VGGNet

K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, ICLR 2015
Comparison: AlexNet vs. VGGNet

- **Receptive fields in the first layer**
  - AlexNet: $11 \times 11$, stride 4
  - Zeiler & Fergus: $7 \times 7$, stride 2
  - VGGNet: $3 \times 3$, stride 1

- **Why that?**
  - If you stack three $3 \times 3$ on top of another $3 \times 3$ layer, you effectively get a $5 \times 5$ receptive field.
  - With three $3 \times 3$ layers, the receptive field is already $7 \times 7$.
  - But much fewer parameters: $3 \cdot 3^2 = 27$ instead of $7^2 = 49$.
  - In addition, non-linearities in-between $3 \times 3$ layers for additional discriminativity.
CNN Architectures: GoogLeNet (2014)

- **Main ideas**
  - “Inception” module as modular component
  - Learns filters at several scales within each module


Image source: Szegedy et al.
GoogLeNet Visualization

Inception module + copies

Auxiliary classification outputs for training the lower layers (deprecated)
## Results on ILSVRC

<table>
<thead>
<tr>
<th>Method</th>
<th>top-1 val. error (%)</th>
<th>top-5 val. error (%)</th>
<th>top-5 test error (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>VGG (2 nets, multi-crop &amp; dense eval.)</td>
<td>23.7</td>
<td>6.8</td>
<td>6.8</td>
</tr>
<tr>
<td>VGG (1 net, multi-crop &amp; dense eval.)</td>
<td>24.4</td>
<td>7.1</td>
<td>7.0</td>
</tr>
<tr>
<td>VGG (ILSVRC submission, 7 nets, dense eval.)</td>
<td>24.7</td>
<td>7.5</td>
<td>7.3</td>
</tr>
<tr>
<td>GoogLeNet (Szegedy et al., 2014) (1 net)</td>
<td>-</td>
<td>-</td>
<td>7.9</td>
</tr>
<tr>
<td>GoogLeNet (Szegedy et al., 2014) (7 nets)</td>
<td>-</td>
<td>-</td>
<td>6.7</td>
</tr>
<tr>
<td>MSRA (He et al., 2014) (11 nets)</td>
<td>-</td>
<td>-</td>
<td>8.1</td>
</tr>
<tr>
<td>MSRA (He et al., 2014) (1 net)</td>
<td>27.9</td>
<td>9.1</td>
<td>9.1</td>
</tr>
<tr>
<td>Clarifai (Russakovsky et al., 2014) (multiple nets)</td>
<td>-</td>
<td>-</td>
<td>11.7</td>
</tr>
<tr>
<td>Clarifai (Russakovsky et al., 2014) (1 net)</td>
<td>-</td>
<td>-</td>
<td>12.5</td>
</tr>
<tr>
<td>Zeiler &amp; Fergus (Zeiler &amp; Fergus, 2013) (6 nets)</td>
<td>36.0</td>
<td>14.7</td>
<td>14.8</td>
</tr>
<tr>
<td>Zeiler &amp; Fergus (Zeiler &amp; Fergus, 2013) (1 net)</td>
<td>37.5</td>
<td>16.0</td>
<td>16.1</td>
</tr>
<tr>
<td>OverFeat (Sermanet et al., 2014) (7 nets)</td>
<td>34.0</td>
<td>13.2</td>
<td>13.6</td>
</tr>
<tr>
<td>OverFeat (Sermanet et al., 2014) (1 net)</td>
<td>35.7</td>
<td>14.2</td>
<td>-</td>
</tr>
<tr>
<td>Krizhevsky et al. (Krizhevsky et al., 2012) (5 nets)</td>
<td>38.1</td>
<td>16.4</td>
<td>16.4</td>
</tr>
<tr>
<td>Krizhevsky et al. (Krizhevsky et al., 2012) (1 net)</td>
<td>40.7</td>
<td>18.2</td>
<td>-</td>
</tr>
</tbody>
</table>

- **VGGNet and GoogLeNet perform at similar level**
  - **Comparison: human performance ~5%** [Karpathy]

http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/
Understanding the ILSVRC Challenge

• Imagine the scope of the problem!
  - 1000 categories
  - 1.2M training images
  - 50k validation images

• This means...
  - Speaking out the list of category names at 1 word/s...
    ...takes 15mins.
  - Watching a slideshow of the validation images at 2s/image...
    ...takes a full day (24h+).
  - Watching a slideshow of the training images at 2s/image...
    ...takes a full month.
More Finegrained Classes

PASCAL
- bird
- cat
- dog

ILSVRC
- flamingo
- cock
- ruffed grouse
- quail
- partridge

- Egyptian cat
- Persian cat
- Siamese cat
- tabby
- lynx

- dalmatian
- keeshond
- miniature schnauzer
- standard schnauzer
- giant schnauzer
Quirks and Limitations of the Data Set

- Generated from WordNet ontology
  - Some animal categories are overrepresented
  - E.g., 120 subcategories of dog breeds

⇒ 6.7% top-5 error looks all the more impressive
Topics of This Lecture

• Recap: CNNs

• CNN Architectures
  ➢ LeNet
  ➢ AlexNet
  ➢ VGGNet
  ➢ GoogLeNet

• Visualizing CNNs
  ➢ Visualizing CNN features
  ➢ Visualizing responses
  ➢ Visualizing learned structures

• Applications
Visualizing CNNs

DeconvNet

ConvNet

Image source: M. Zeiler, R. Fergus
Visualizing CNNs


Image source: M. Zeiler, R. Fergus
Visualizing CNNs

Layer 3

Image source: M. Zeiler, R. Fergus
Visualizing CNNs

Layer 4

Layer 5

Image source: M. Zeiler, R. Fergus
What Does the Network React To?

• Occlusion Experiment
  - Mask part of the image with an occluding square.
  - Monitor the output
What Does the Network React To?

Input image

p(True class)

Most probable class

Image source: M. Zeiler, R. Fergus

Slide credit: Svetlana Lazebnik, Rob Fergus
What Does the Network React To?

Input image

- Total activation in most active 5\textsuperscript{th} layer feature map
- Other activations from the same feature map.

Image source: M. Zeiler, R. Fergus

Slide credit: Svetlana Lazebnik, Rob Fergus
What Does the Network React To?

Input image

\[ p(\text{True class}) \]

Most probable class

Image source: M. Zeiler, R. Fergus

Slide credit: Svetlana Lazebnik, Rob Fergus
What Does the Network React To?

Input image

Total activation in most active 5th layer feature map

Other activations from the same feature map.

Image source: M. Zeiler, R. Fergus

Slide credit: Svetlana Lazebnik, Rob Fergus
What Does the Network React To?

Input image

p(True class)  Most probable class

Image source: M. Zeiler, R. Fergus

Slide credit: Svetlana Lazebnik, Rob Fergus
What Does the Network React To?

Input image

Total activation in most active 5th layer feature map

Other activations from the same feature map.

Image source: M. Zeiler, R. Fergus
Slide credit: Svetlana Lazebnik, Rob Fergus
Inceptionism: Dreaming ConvNets

- **Idea**
  - Start with a random noise image.
  - Enhance the input image such as to enforce a particular response (e.g., banana).
  - Combine with prior constraint that image should have similar statistics as natural images.
  
  \[ \Rightarrow \text{Network hallucinates characteristics of the learned class.} \]

http://googleresearch.blogspot.de/2015/06/inceptionism-going-deeper-into-neural.html
Inceptionism: Dreaming ConvNets

• Results

http://googleresearch.blogspot.de/2015/07/deepdream-code-example-for-visualizing.html
Inceptionism: Dreaming ConvNets

https://www.youtube.com/watch?v=IREsx-xWQ0g
Topics of This Lecture

• Recap: CNNs

• CNN Architectures
  - LeNet
  - AlexNet
  - VGGNet
  - GoogLeNet

• Visualizing CNNs
  - Visualizing CNN features
  - Visualizing responses
  - Visualizing learned structures

• Applications
The Learned Features are Generic

- Experiment: feature transfer
  - Train network on ImageNet
  - Chop off last layer and train classification layer on CalTech256
  - State of the art accuracy already with only 6 training images

Image source: M. Zeiler, R. Fergus
Other Tasks: Detection

R-CNN: *Regions with CNN features*

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

- Results on PASCAL VOC Detection benchmark
  - Pre-CNN state of the art: 35.1% mAP [Uijlings et al., 2013]
  - 33.4% mAP DPM
  - R-CNN: 53.7% mAP

Other Tasks: Semantic Segmentation

[Farabet et al. ICML 2012, PAMI 2013]
Other Tasks: Semantic Segmentation

[Farabet et al. ICML 2012, PAMI 2013]
Other Tasks: Face Verification

Y. Taigman, M. Yang, M. Ranzato, L. Wolf, DeepFace: Closing the Gap to Human-Level Performance in Face Verification, CVPR 2014

Slide credit: Svetlana Lazebnik
Commercial Recognition Services

- E.g., clarifai

Upload an image or video file under 100mb or give us a direct link to a file on the web.

- Be careful when taking test images from Google Search
  - Chances are they may have been seen in the training set...

Image source: clarifai.com
Commercial Recognition Services

B. Leibe

Image source: clarifai.com
References and Further Reading

- **LeNet**

- **AlexNet**

- **VGGNet**

- **GoogLeNet**
Effect of Multiple Convolution Layers

Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Slide credit: Yann LeCun