Lecture 11:

Case Studies
LeNet – AlexNet – ZFNet – VGG
Inception – Resnet

Alireza Akhavan Pour
CLASS.VISION
Convolutional Neural Networks

[LeNet-5, LeCun 1980]
VIEW ON THE VISUAL PATHWAY

BEHAVIOR OF AN GANGLION CELL

On center

No light

Light on

No light

Off center

No light

Light on

No light
VIEW ON THE VISUAL PATHWAY, SIMPLE CORTICAL CELLS

- No light
- Wrong position-wrong orientation
- Wright position-wrong orientation
- Wrong position-wright orientation
- Wright position Wright orientation
Behavior of a complex cell...
A bit of history:

Hubel & Wiesel,
1959
RECEPTIVE FIELDS OF SINGLE NEURONES IN THE CAT'S STRIATE CORTEX
1962
RECEPTIVE FIELDS, BINOCULAR INTERACTION AND FUNCTIONAL ARCHITECTURE IN THE CAT'S VISUAL CORTEX
1968...
یافته‌های Wiesel و Hubel:

- ساختار سلسله مراتبی
- افزایش انتخاب پذیری نسبت به اشیا پیچیده
- افزایش بازشناسی نامتغیر برای تغییرات اشیا (receptive field)
- افزایش اندازه میدان پذیرندگی (feed-forward)
- بازشناسایی بلافاصله: جلو رونده
A bit of history

Topographical mapping in the cortex: nearby cells in cortex represented nearby regions in the visual field
A bit of history

Topographical mapping in the cortex: nearby cells in cortex represent nearby regions in the visual field
Hierarchical organization

Hubel & Weisel

- Topographical mapping

- Simple cells
- Complex cells
- Hyper-complex cells

Featural hierarchy

- High level
- Mid level
- Low level
Hierarchical organization

Simple cells: Response to light orientation

Complex cells: Response to light orientation and movement

Hypercomplex cells: Response to movement with an end point

Illustration of hierarchical organization in early visual pathways by Lane McIntosh, copyright CS231n 2017
A bit of history:

Neurocognitron

[Fukushima 1980]

“sandwich” architecture (SCSCSC...)

simple cells: modifiable parameters

complex cells: perform pooling
A bit of history:
Gradient-based learning applied to document recognition
[LeCun, Bottou, Bengio, Haffner 1998]

LeNet-5
Demo

https://www.youtube.com/watch?v=AqkflQ4IGaM
Classic networks:

- LeNet-5
- AlexNet
- ZFNet
- VGG

ResNet

Inception
LeNet
LeNet - 5

[LeCun et al., 1998. Gradient-based learning applied to document recognition]
AlexNet
AlexNet

227×227×3

3×3 same
13×13×384

55×55×96

3×3 s = 4
11×11

27×27×96

3×3 s = 2

27×27×256

5×5 same

27×27×256

3×3 s = 2

13×13×256

MAX-POOL

3×3 same

3×3 s = 2

3×3 s = 2

9216

13×13×384

3×3

13×13×384

3×3

13×13×256

6×6×256

5×5

5×5

5×5

1000

MAX-POOL

MAX-POOL

MAX-POOL

Softmax

⋮

Softmax

Krizhevsky et al., 2012. ImageNet classification with deep convolutional neural networks]
Case Study: AlexNet

[Krizhevsky et al. 2012]

Input: 227x227x3 images

**First layer** (CONV1): 96 11x11 filters applied at stride 4

=>

Output volume [55x55x96]

**Q: What is the total number of parameters in this layer?**
Case Study: AlexNet

[Krizhevsky et al. 2012]

Input: 227x227x3 images

First layer (CONV1): 96 11x11 filters applied at stride 4
=>
Output volume [55x55x96]
Parameters: (11*11*3)*96 = 35K
Case Study: AlexNet

[Krizhevsky et al. 2012]

Input: 227x227x3 images
After CONV1: 55x55x96

Second layer (POOL1): 3x3 filters applied at stride 2

Q: what is the output volume size? Hint: (55-3)/2+1 = 27
Case Study: AlexNet

[Krizhevsky et al. 2012]

Input: 227x227x3 images
After CONV1: 55x55x96

Second layer (POOL1): 3x3 filters applied at stride 2
Output volume: 27x27x96

Q: what is the number of parameters in this layer?
Case Study: AlexNet

[Krizhevsky et al. 2012]

Input: 227x227x3 images
After CONV1: 55x55x96

Second layer (POOL1): 3x3 filters applied at stride 2
Output volume: 27x27x96
Parameters: 0!
Case Study: AlexNet

[Krizhevsky et al. 2012]

Full (simplified) AlexNet architecture:

[227x227x3] INPUT

[55x55x96] CONV1: 96 11x11 filters at stride 4, pad 0

[27x27x96] MAX POOL1: 3x3 filters at stride 2

[27x27x96] NORM1: Normalization layer

[27x27x256] CONV2: 256 5x5 filters at stride 1, pad 2

[13x13x256] MAX POOL2: 3x3 filters at stride 2

[13x13x256] NORM2: Normalization layer

[13x13x384] CONV3: 384 3x3 filters at stride 1, pad 1

[13x13x384] CONV4: 384 3x3 filters at stride 1, pad 1

[13x13x256] CONV5: 256 3x3 filters at stride 1, pad 1

[6x6x256] MAX POOL3: 3x3 filters at stride 2

[4096] FC6: 4096 neurons

[4096] FC7: 4096 neurons

[1000] FC8: 1000 neurons (class scores)

Details/Retrospectives:
- first use of ReLU
- used Norm layers (not common anymore)
- heavy data augmentation
- dropout 0.5
- batch size 128
- SGD Momentum 0.9
- Learning rate 1e-2, reduced by 10 manually when val accuracy plateaus
- L2 weight decay 5e-4
- 7 CNN ensemble: 18.2% -> 15.4%
ZFNet
Case Study: ZFNet

ImageNet top 5 error: 15.4% -> 14.8%

AlexNet but:
CONV1: change from (11x11 stride 4) to (7x7 stride 2)
CONV3,4,5: instead of 384, 384, 256 filters use 512, 1024, 512

[Zeiler and Fergus, 2013]
VGG
VGG - 16

CONV = 3×3 filter, s = 1, same    MAX-POOL = 2×2, s = 2

(Conv 64) x2

[CONV 224x224x64]

112x112 x64

(Conv 128) x2

112x112 x128

POOL

56x56 x128

(Conv 256) x3

28x28 x256

(Conv 512) x3

28x28 x512

POOL

14x14 x512

(Conv 512) x3

14x14 x512

POOL

7x7 x512

FC 4096

FC 4096

Softmax

1000

Case Study: VGGNet
[Simonyan and Zisserman, 2014]

Only 3x3 CONV stride 1, pad 1
and 2x2 MAX POOL stride 2

TOTAL memory: 24M * 4 bytes ~ 93MB / image
(only forward! ~*2 for bwd)
TOTAL params: 138M parameters

best model

11.2% top 5 error in ILSVRC 2013 -> 7.3% top 5 error
Classic networks:

- LeNet-5
- AlexNet
- ZFNet
- VGG

7 CNN ensemble: 15.4% top 5 error
14.8% top 5 error
7.3% top 5 error

Inception

ResNet
One by One Convolution!
Why does a $1 \times 1$ convolution do?

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>6</th>
<th>5</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>9</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>8</td>
<td>5</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>3</td>
<td>7</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>9</td>
<td>8</td>
<td>3</td>
<td>5</td>
</tr>
</tbody>
</table>

$6 \times 6$

$6 \times 6 \times 32$

\[ * \]

$2$

\[ = \]

$2 \times 4 \times 6$

$6 \times 6 \times \# \text{ filters}$

[Lin et al., 2013. Network in network]
[Lin et al., 2013. Network in network]
Inception

(GoogLeNet)
WE NEED TO GO DEEPER

http://knowyourmeme.com/memes/we-need-to-go-deeper
[Szegedy et al. 2014. Going deeper with convolutions]
Inception module

Previous Activation

28 x 28 x 192

1 x 1 CONV

28 x 28 x 64

1 x 1 CONV

28 x 28 x 128

1 x 1 CONV

28 x 28 x 32

5 x 5 CONV

28 x 28 x 32

3 x 3, s = 1

same

SRTTU – A.Akhavan

Lecture 11 - 38
Inception network

[Szegedy et al., 2014, Going Deeper with Convolutions]
Case Study: GoogLeNet

Fun features:
- Only 5 million params! (Removes FC layers completely)

Compared to AlexNet:
- 12X less params
- 2x more compute
- 6.67% (vs. 16.4%)
Case Study: GoogLeNet

Inception module

ILSVRC 2014 winner (6.7% top 5 error)

[Szegedy et al., 2014, Going Deeper with Convolutions]
ResNet
Residual block

\[ a^l \xrightarrow{\text{Linear}} \text{ReLU} a^{l+1} \xrightarrow{\text{Linear}} \text{ReLU} \rightarrow a^{l+2} \]

\[ z^{l+1} = W^{l+1} a^l + b^{l+1} \quad a^{l+1} = g(z^{l+1}) \quad z^{l+2} = W^{l+2} a^{l+1} + b^{l+2} \quad a^{l+2} = g(z^{l+2}) \]

\[ a^{l+2} = g(z^{l+2} + a^l) \]

[He et al., 2015. Deep residual networks for image recognition]
Residual Network

ResNet

Plain

[He et al., 2015. Deep residual networks for image recognition]
Case Study: ResNet [He et al., 2015]

MSRA @ ILSVRC & COCO 2015 Competitions

- **1st places in all five main tracks**
  - ImageNet Classification: “Ultra-deep” (quote Yann) **152-layer** nets
  - ImageNet Detection: **16%** better than 2nd
  - ImageNet Localization: **27%** better than 2nd
  - COCO Detection: **11%** better than 2nd
  - COCO Segmentation: **12%** better than 2nd

*improvements are relative numbers


Slide from Kaiming He’s recent presentation

https://www.youtube.com/watch?v=1PGLj-uKT1w
Case Study: ResNet [He et al., 2015]

ILSVRC 2015 winner (3.6% top 5 error)

Revolution of Depth

- AlexNet, 8 layers (ILSVRC 2012)
- VGG, 19 layers (ILSVRC 2014)
- ResNet, 152 layers (ILSVRC 2015)

2-3 weeks of training on 8 GPU machine

at runtime: faster than a VGGNet! (even though it has 8x more layers)

(slide from Kaiming He’s recent presentation)
Case Study: ResNet

[He et al., 2015]

Spatial dimension only 56x56!
Revolution of Depth

152 layers

ImageNet Classification top-5 error (%)

ILSVRC'15 ResNet: 3.57
ILSVRC'14 GoogleNet: 6.7
ILSVRC'14 VGG: 7.3
ILSVRC'13: 11.7
ILSVRC'12 AlexNet: 16.4
ILSVRC'11: 25.8
ILSVRC'10: 28.2

(slides from Kaiming He’s recent presentation)
منابع

• https://www.slideshare.net/Alirezaakhavanpour/akhavan-2
• https://www.coursera.org/specializations/deep-learning
• http://cs231n.stanford.edu/