Convolutional neural nets

(and the brain)

CISC 5800

Extra content

Innovations in computer vision: Convolutional neural networks

- Introduced by Yann LeCun (IEEE 1998) for digit recognition
- Popularized by Alex Krizhevsky (NIPS 2012) for broad object recognition

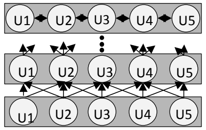
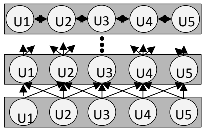


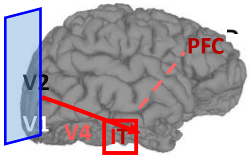
Image-Net: photos of >100K object classes

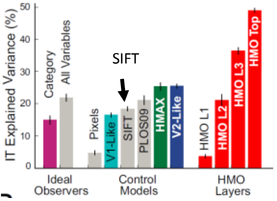
2012: best non conv-net 26% error rate

Year	Group	Error
2012	Krizhevsky	15.3%
2014	VGG	7.3%
2014	GoogLeNet	6.7%
200,000BC	Human Vision	5.1%

Computer models of cortical vision 2.0

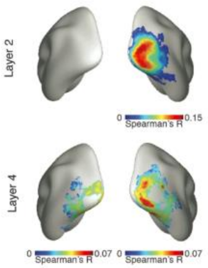






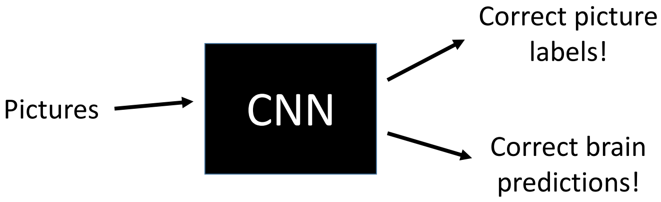
Yamins PNAS 2014

Optimized CNNs



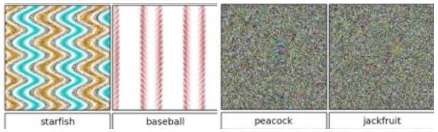
Cichy Nature 2016

What are CNNs?



Limits: Fooling CNNs

Nguyen CVPR 2015

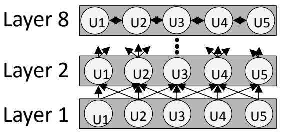


Why understand CNNs?

Insights on:

- Making better-performing models
- Making simpler models
- How the brain actually works

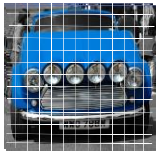
How do CNNs work?



Collection of “neurons” divided among *k* layers

Each neuron looks for one pattern

Each neuron looks for same pattern at multiple locations in input



U1

10	1	40	0
0	3	65	15
0	12	12	0
0	5	15	0

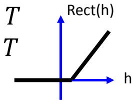
U2

0	0	10	25
0	90	0	6
0	40	25	0
0	14	0	0

Cascade of linear and non-linear computations

Summation $f(\boldsymbol{x}) = \sum_i w_i x_i$

Rectification $g(y) = \begin{cases} 0 & y \le T \\ y - T & y > T \end{cases}$

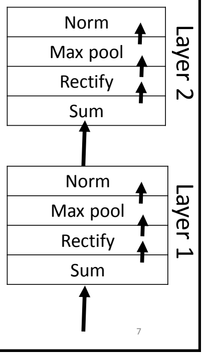


Max pool $h(\boldsymbol{z}) = \max(z_1, \cdots, z_n)$

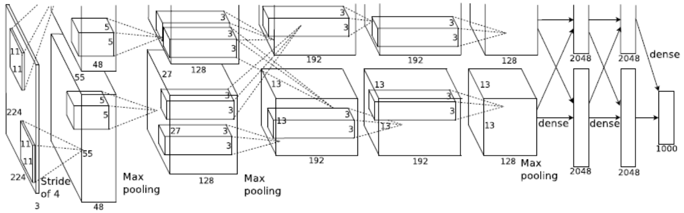
10	1	40
0	3	65
0	12	12

→ 65

Normalization $\widetilde{r_{x,y}} = \frac{r_{x,y}}{k+\sum r_{x,y}^2}$



Example full network – Krizhevsky NIPS 2012



Eight layers One, two, or four sub-layers

256 – 384 neurons per layer