Convolutional neural nets

CISC 5800
Professor Daniel Leeds

Innovations in computer vision: Convolutional neural networks

- Introduced by Yann LeCun (IEEE 1988) for digit recognition
- Popularized by Alex Krizhevsky (NIPS 2012) for broad object recognition

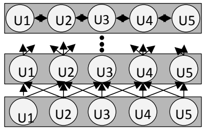


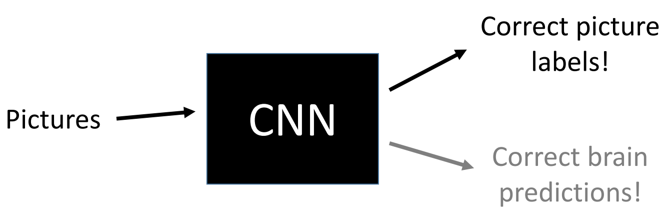
Image-Net: photos of >100K object classes
2012: best non conv-net 26% error rate

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

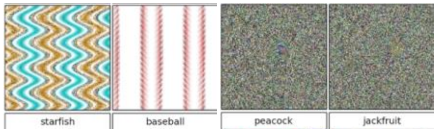
Top 5 - % data where top 5 most likely classes were NOT correct class

2

What are CNNs?



Limits: Fooling CNNs
Nguyen CVPR 2015



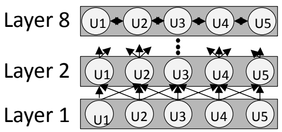
4

Why understand CNNs?

Insights on:
- Making better-performing models
- Making simpler models
- How the brain actually works

5

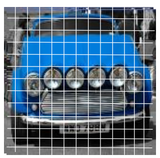
How do CNNs work?



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

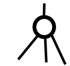
U2

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

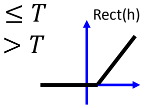
6

Cascade of linear and non-linear computations

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



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

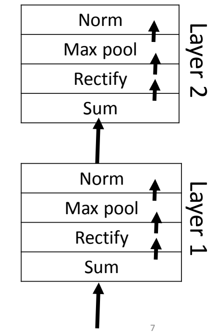


Max pool $h(\boldsymbol{z}) = \max(z_1, \cdots, z_n)$

10	1	40
0	3	65
0	12	12

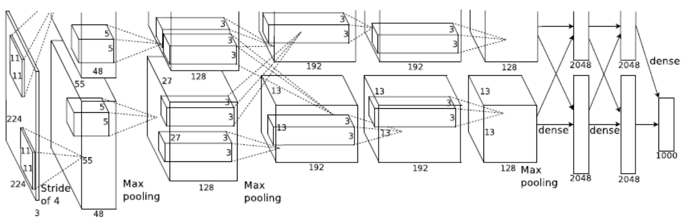
$\rightarrow \boxed{65}$

Normalization $\widetilde{r_{x,y}} = \dfrac{r_{x,y}}{\left(k+\alpha \sum_j {r^j_{x,y}}^2\right)^\beta}$



7

Example full network – Krizhevsky NIPS 2012



Eight layers

One, two, or four sub-layers

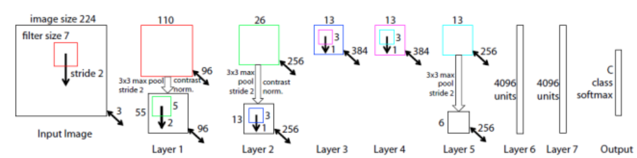
256 – 384 neurons per layer

8

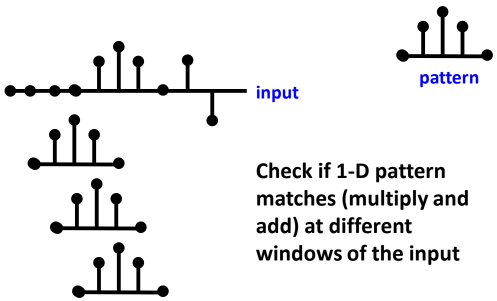
Convolution

Each neuron looks for same pattern at multiple locations in input

- How big a location (size)?
- How many locations (stride)?

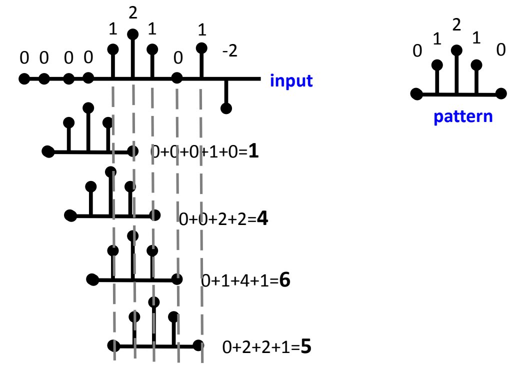


1D convolution



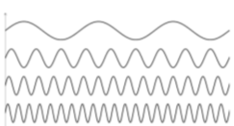
Check if 1-D pattern matches (multiply and add) at different windows of the input

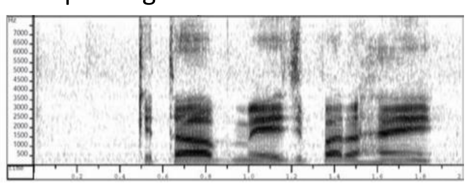
1D convolution



"Spectrogram" as image

- Speech, motion, stock-prices converted to frequency-over-time
- Learn 2D patterns from spectrograms



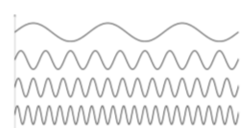


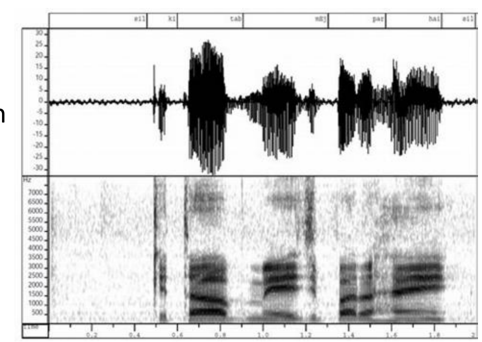
- Or learn wave-gram from wavelets

Speech spectrum: convolution of sine waves

Shifting time windows,

sine waves at each frequency





Inflating data set

- Flip/rotate image

- Change lighting/contrast