

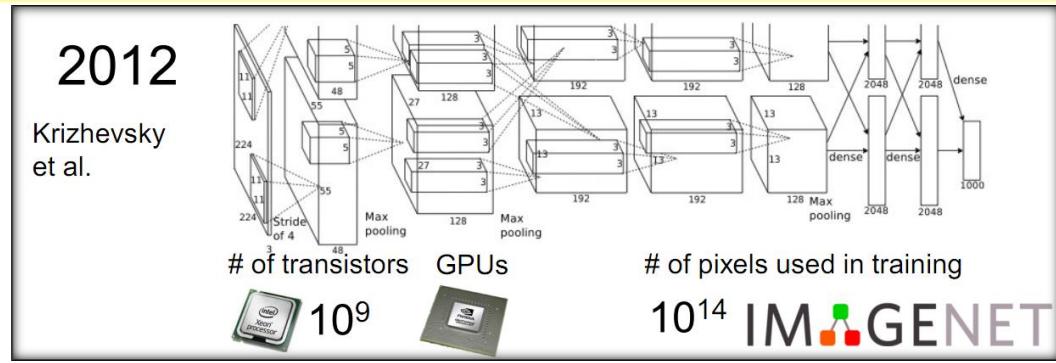
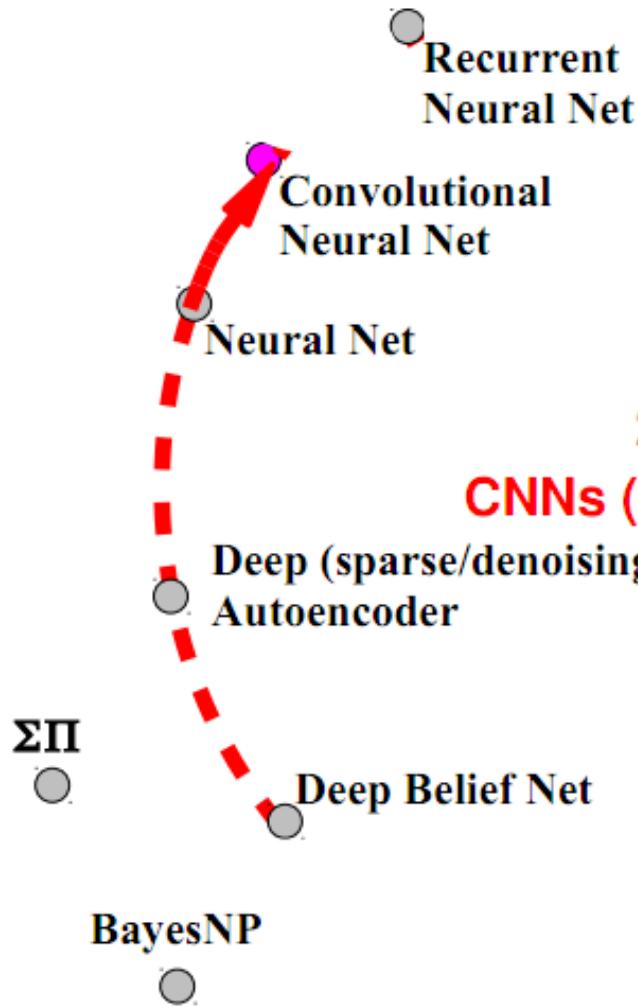
Lecture 7: Intro to CNNs



Dr. Đinh Viết Sang
Hanoi University of Science and Technology

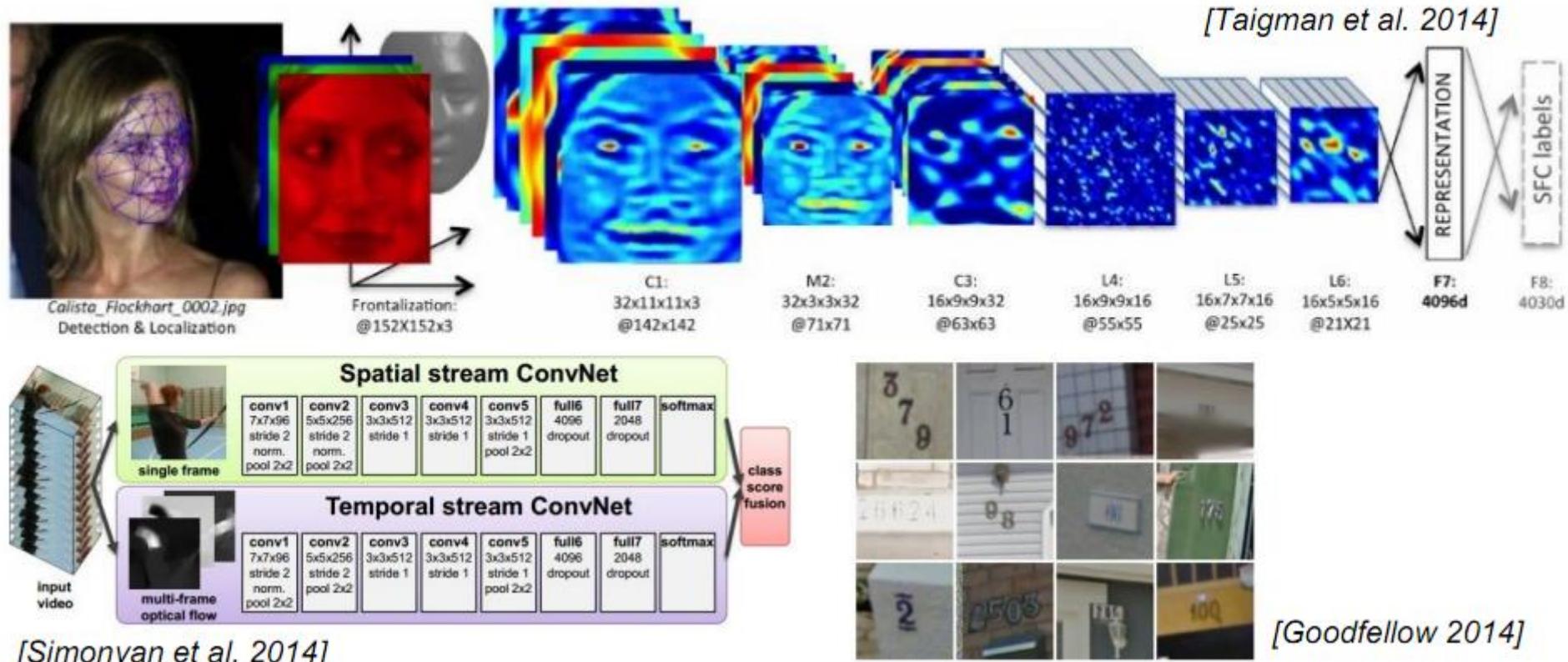
Hanoi 2016

History of DL



ConvNets

Fast-forward to today: ConvNets are everywhere



ConvNets

Fast-forward to today: ConvNets are everywhere

Classification



Retrieval



[Krizhevsky 2012]

ConvNets

Fast-forward to today: ConvNets are everywhere



[Toshev, Szegedy 2014]

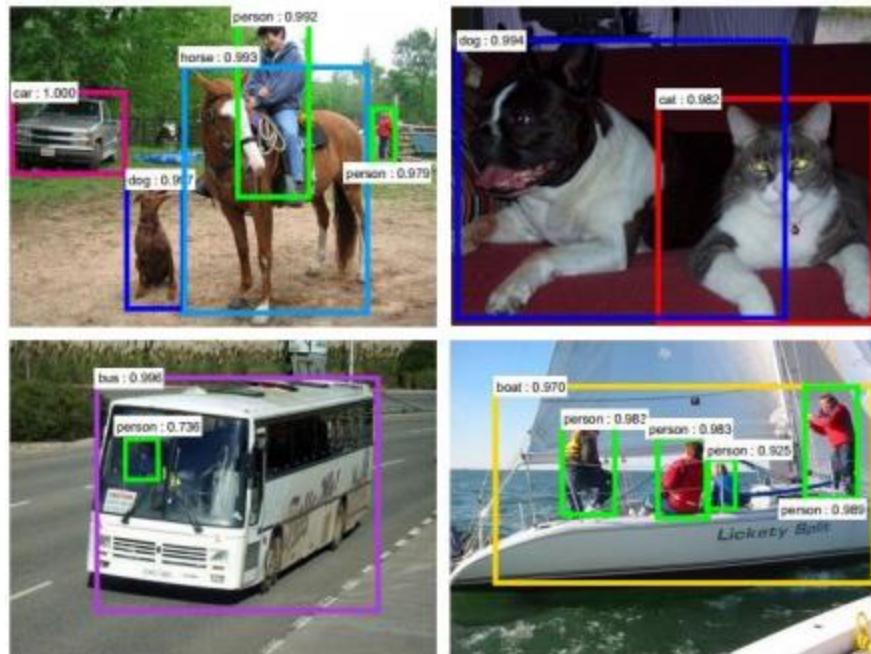


[Mnih 2013]

ConvNets

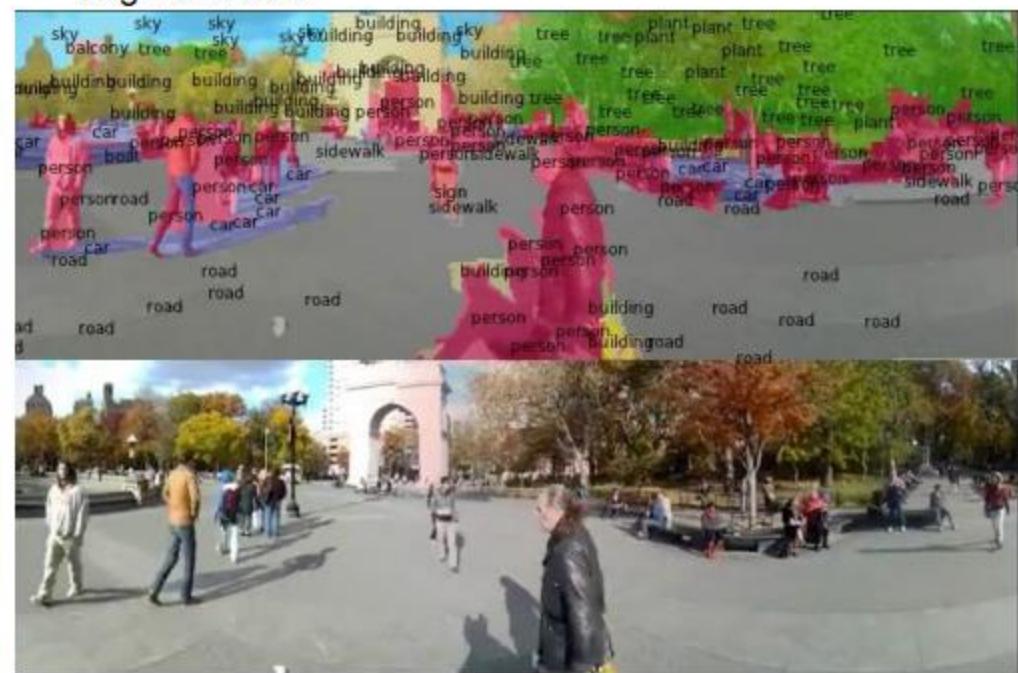
Fast-forward to today: ConvNets are everywhere

Detection



[Faster R-CNN: Ren, He, Girshick, Sun 2015]

Segmentation



[Farabet et al., 2012]

ConvNets

Fast-forward to today: ConvNets are everywhere



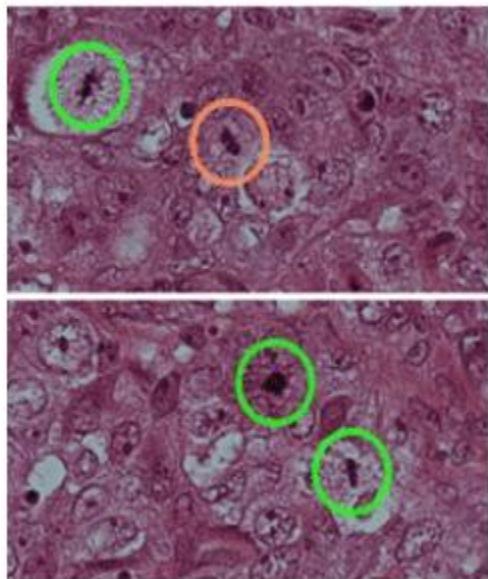
self-driving cars



NVIDIA Tegra X1

ConvNets

Fast-forward to today: ConvNets are everywhere



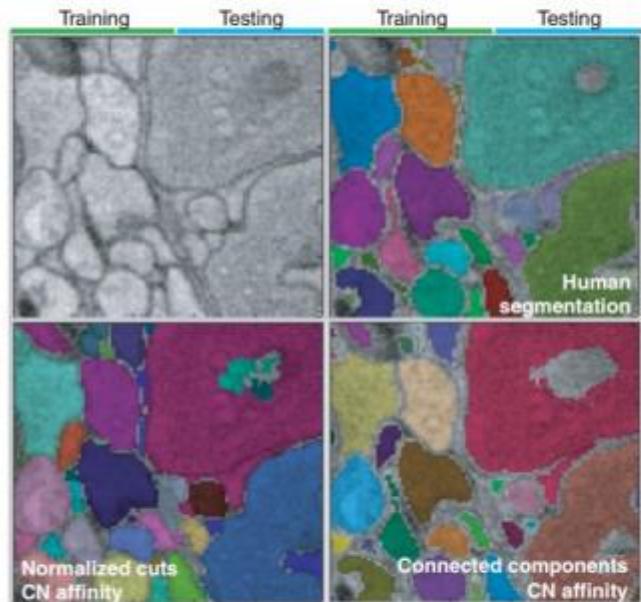
[Ciresan et al. 2013]



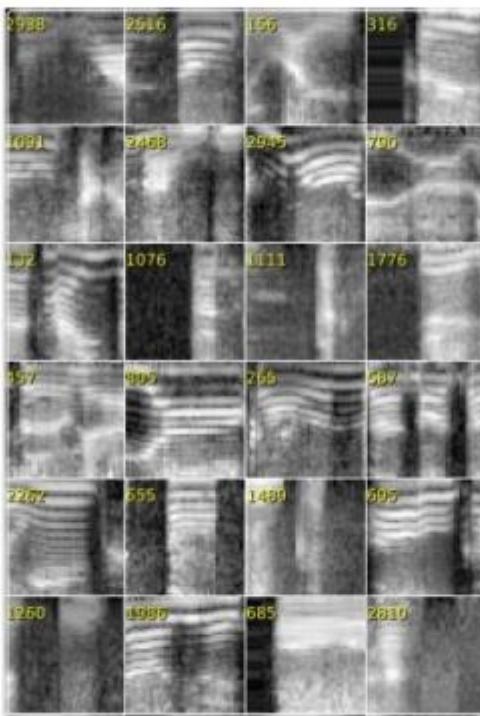
[Sermanet et al. 2011]
[Ciresan et al.]

ConvNets

Fast-forward to today: ConvNets are everywhere



[Turaga et al., 2010]



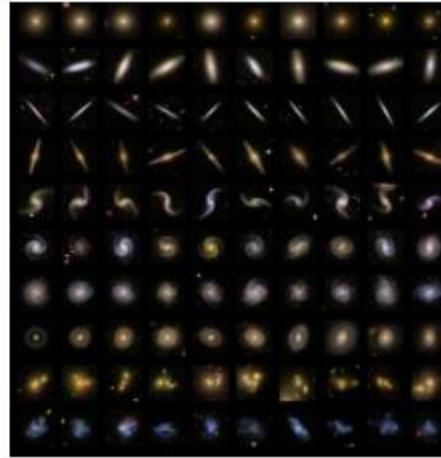
I caught this movie on the Sci-Fi channel recently. It actually turned out to be pretty decent as far as B-list horror/suspense films go. **The guys time naive and we had** **sounded a "911" take a road trip to stop a robbery but have the worst possible luck when a scoundrel in a frosty, make-shift back-track hybrid device** **to play cat-and-mouse with them.** Things are further complicated when they pick up a ridiculously whorish hitchhiker. What makes this film unique is that the combination of comedy and terror actually works in this movie, unlike so many others. The two guys are likable enough and there are some good chase/suspense scenes. Nice pacing and comic timing make this movie more than possible for the home/theater tact. **Definitely worth checking out.**

I just saw this on a local independent station in the New York City area. **The same showed promise but when I saw the director George Cosmatos, I became suspicious. And very strong. It was every bit as bad, every bit as pointless and stupid as every George Cosmatos movie I ever saw.** He's like a stupid man's Michael Bay – with all the awfulness that accolade promises. There's no point to the conspiracy, no burning issues that urge the conspirators on. We are left to ourselves to connect the dots from one bit of graffiti on various walls in the film to the next. Thus, the current budget crisis, the war in Iraq, Islamic extremism, the fate of social security, 47 million Americans without health care, stagnating wages, and the death of the middle class are all subsumed by the sheer force of graffiti. A truly, stunningly shitty film.

Graphics is far from the best part of the game. **This is the number one best TH game in the series. Neu to Underground. It deserves strong love. It is an intense game.** There are massive levels, massive unlockable characters... it's just a massive game. **Waste your money on this game. This is the kind of money that is wasted.** project. And even though graphics suck, that doesn't make a game good. Actually, the graphics were good at the time. Today the graphics are crap. WHO CARES? As they say in Canada. This is the first game, aye. (You got to go to Canada in THPS3) Well, I don't know if they say that, but they might. who knows. Well, Canadian people do. Wait a minute. I'm getting off topic. This game rocks. Buy it, play it, enjoy it, love it. It's PURE BRILLIANCE.

The first was good and original. I was a not bad independently movie. So I heard a second one was made and I had to watch it. What really makes this movie work is Hold Nelson's character and the sometimes clever script. **A pretty good script for a person who wrote the Final Destination films and the direction was okay.** Sometimes there's scenes where it looks like it was filmed using a home video camera with a grain - look. Great made - for - TV movie. **If you worth the rental, and probably worth buying just to get that nice eerie feeling and watch Jack Nelson's Stanley doing what he does best!** I suggest newscasters to watch the first one before watching the sequel, just to you'll have an idea what Stanley is like and get a little history background.

[Denil et al. 2014]



ConvNets

Describes without errors



A person riding a motorcycle on a dirt road.

Describes with minor errors



Two dogs play in the grass.

Somewhat related to the image



A skateboarder does a trick on a ramp.

Unrelated to the image



A dog is jumping to catch a frisbee.

Image Captioning



A group of young people playing a game of frisbee.



Two hockey players are fighting over the puck.



A little girl in a pink hat is blowing bubbles.



A refrigerator filled with lots of food and drinks.



A herd of elephants walking across a dry grass field.



A close up of a cat laying on a couch.



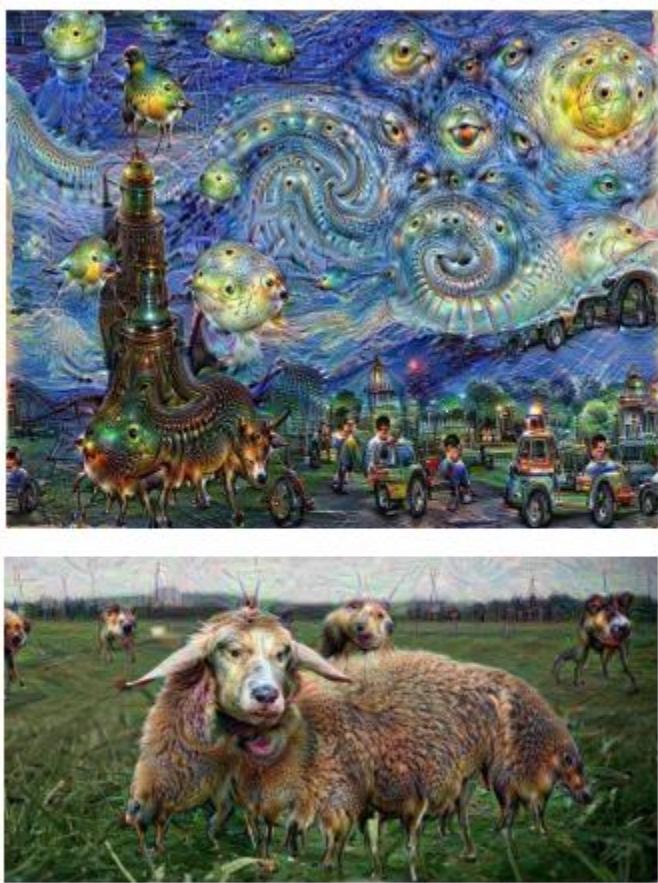
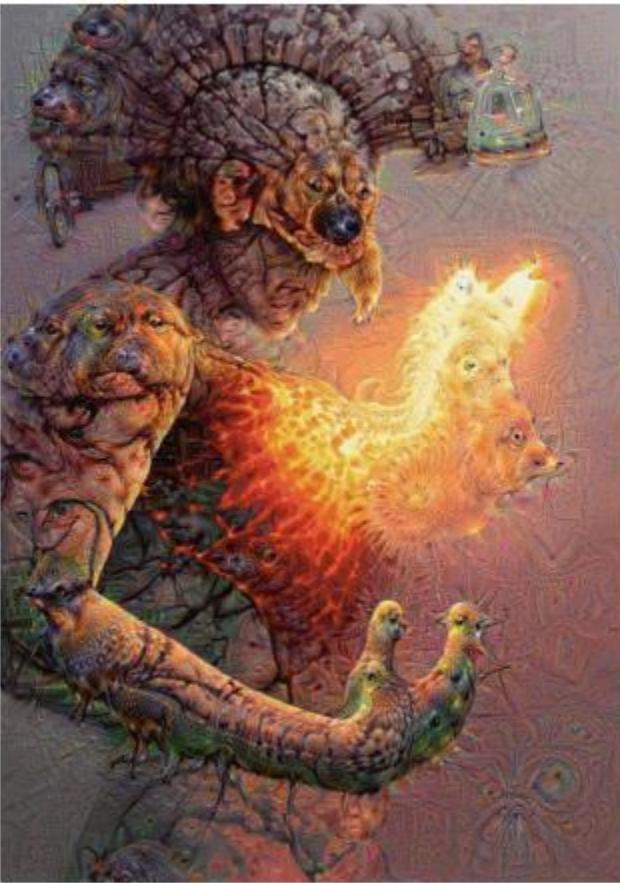
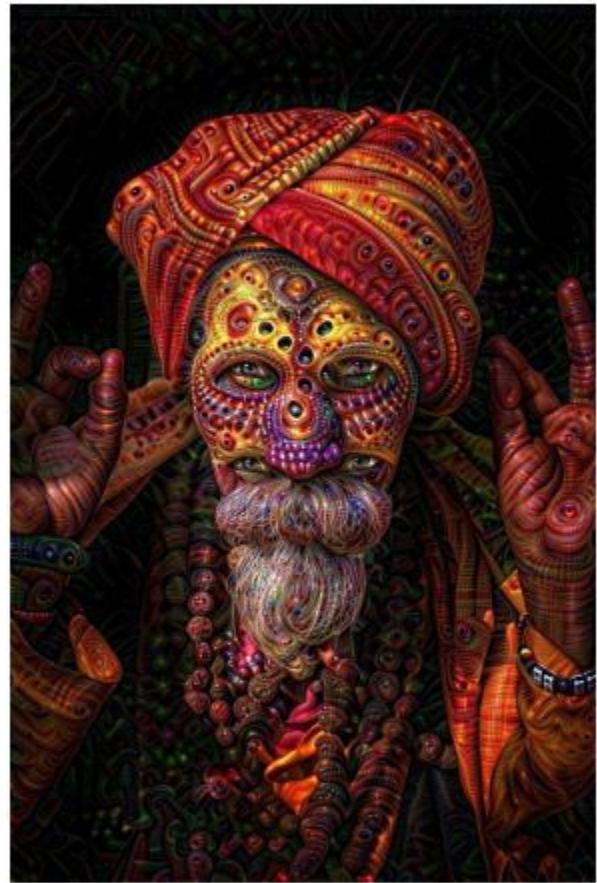
A red motorcycle parked on the side of the road.



A yellow school bus parked in a parking lot.

[Vinyals et al., 2015]

ConvNets



[reddit.com/r/deepdream](https://www.reddit.com/r/deepdream)

Statistical Invariance



CAT!

Statistical Invariance

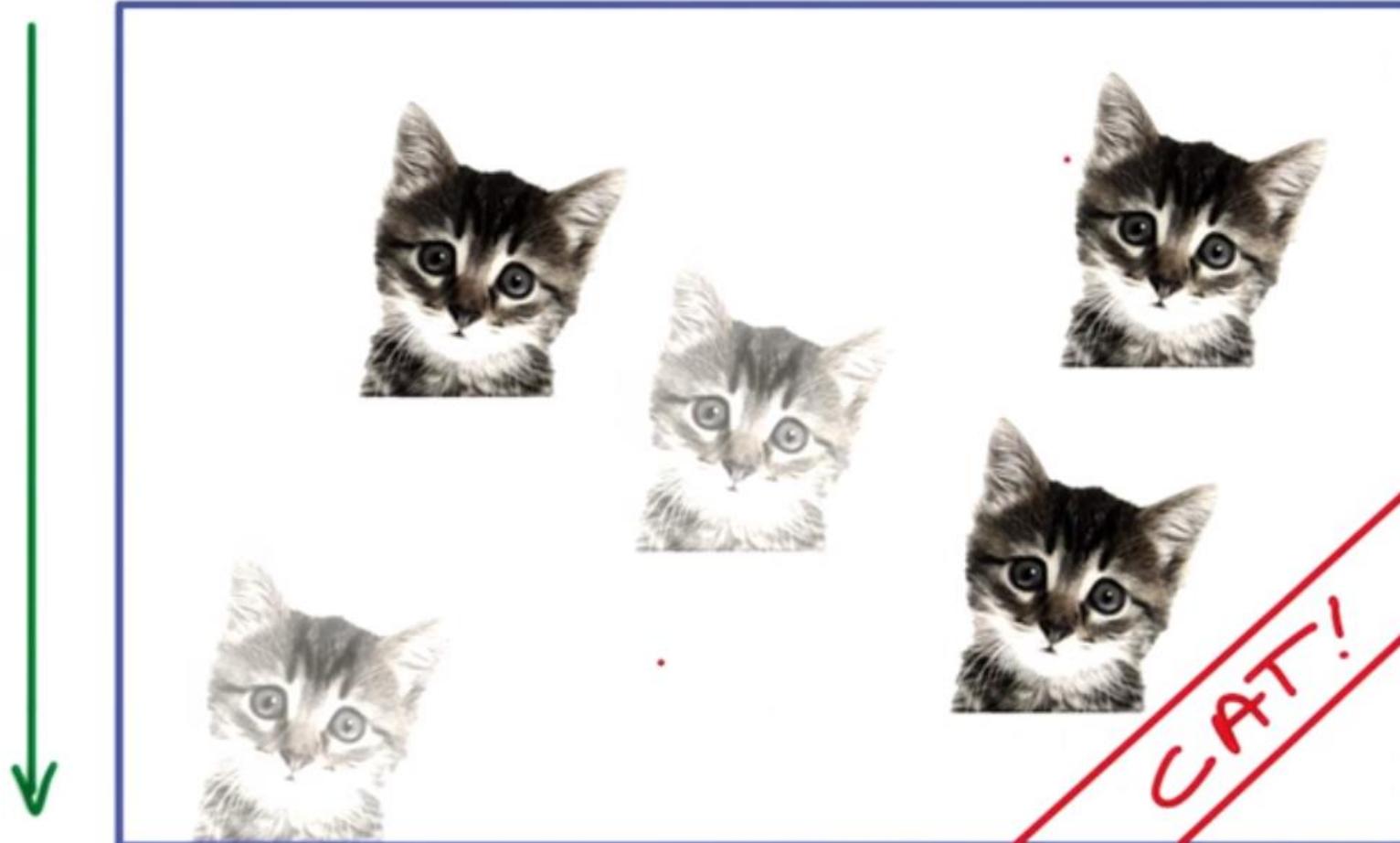


Statistical Invariance

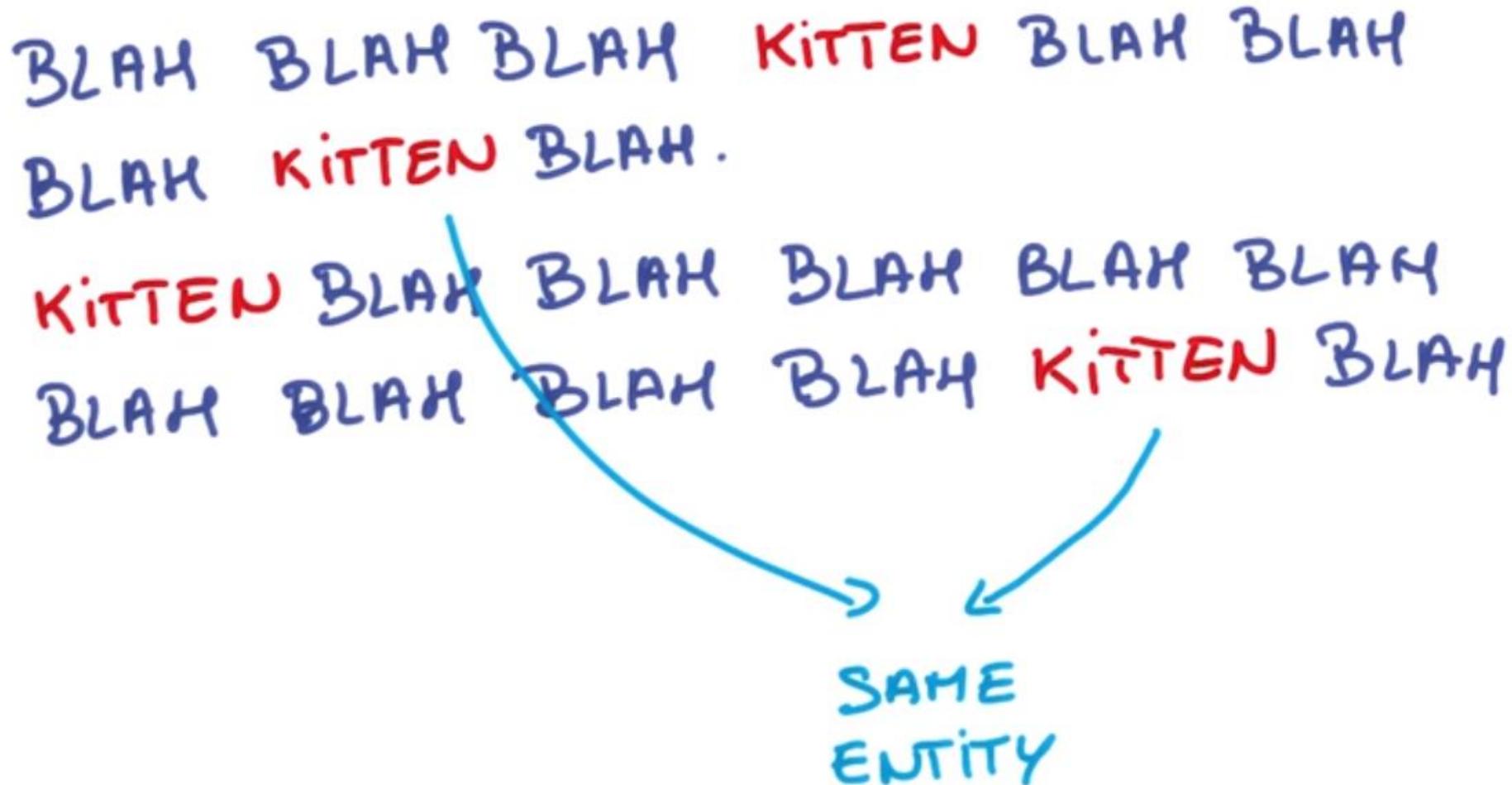


Statistical Invariance

TRANSLATION INVARIANCE →



Statistical Invariance



Statistical Invariance



BLAH BLAH BLAH KITTEN BLAH BLAH
BLAH KITTEN BLAH.
KITTEN BLAH BLAH BLAH BLAH BLAH
BLAH BLAH BLAH BLAH KITTEN BLAH

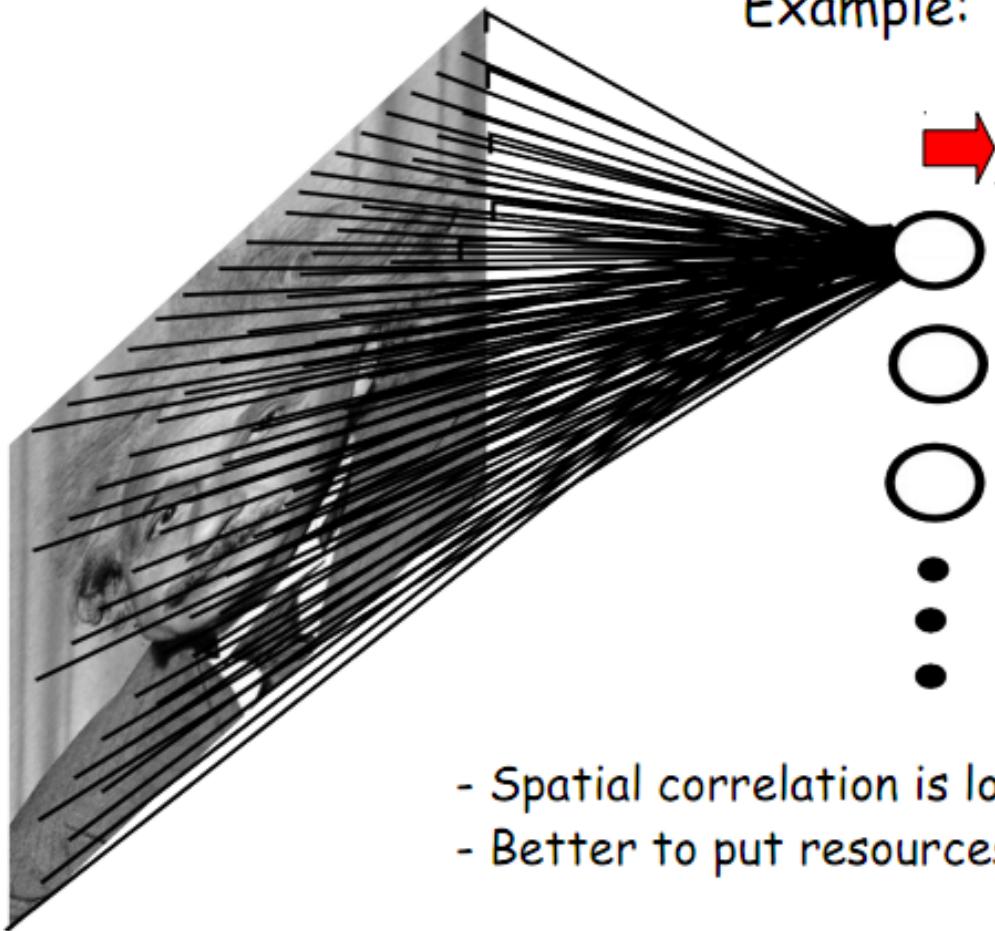
w_3 | w_3 w_3 w_3

w_3

WEIGHT
SHARING

Neural Networks

FULLY CONNECTED NEURAL NET

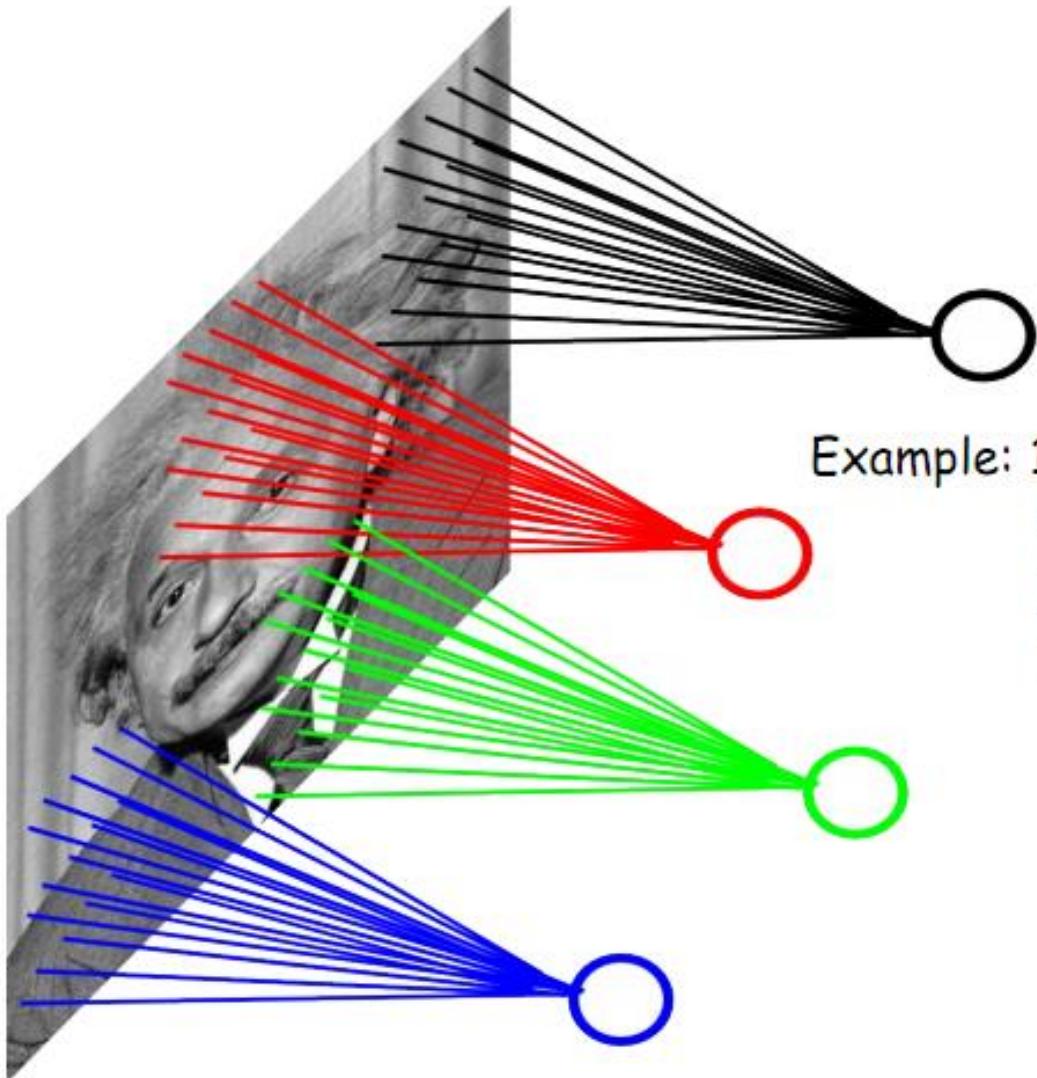


Example: 1000x1000 image
1M hidden units
→ 10^{12} parameters!!!

- Spatial correlation is local
- Better to put resources elsewhere!

Local Connectivity

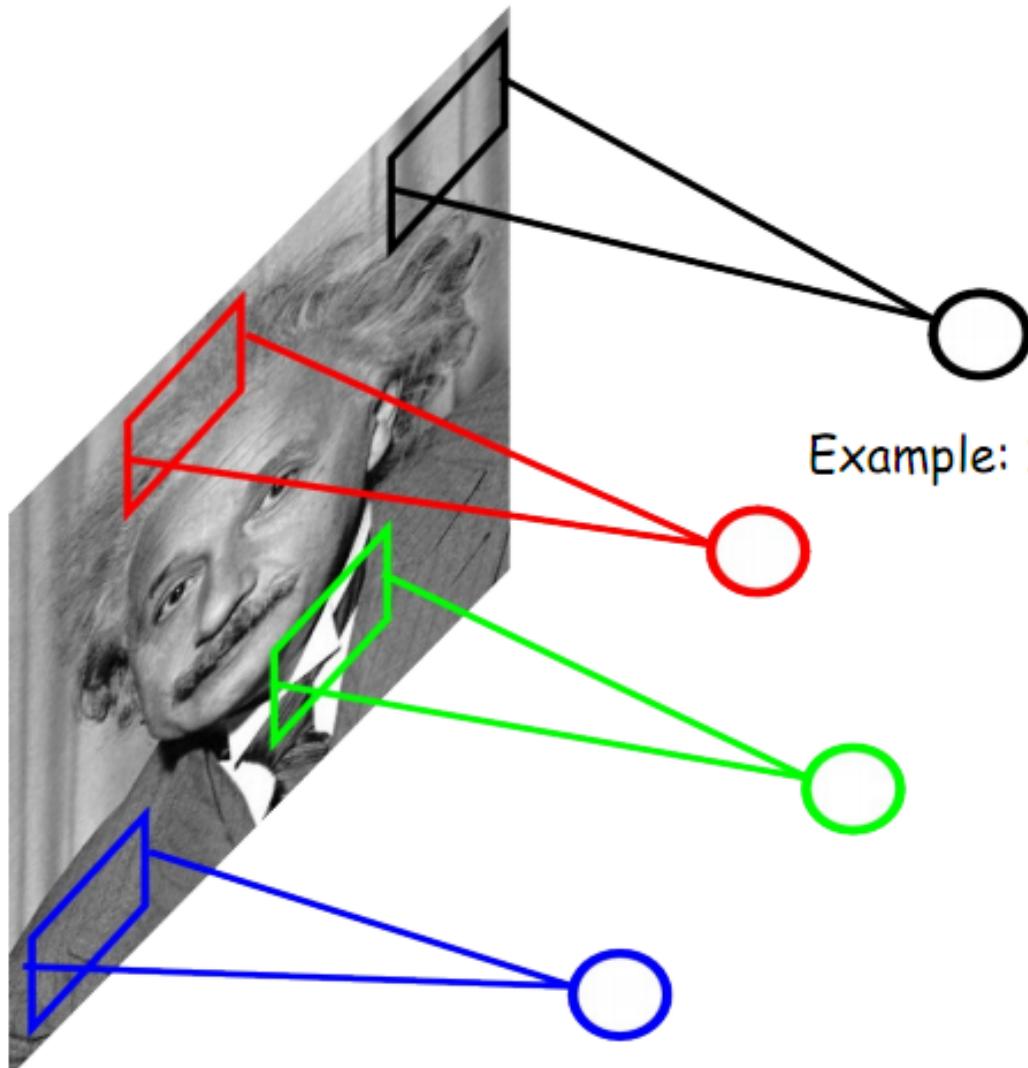
LOCALLY CONNECTED NEURAL NET



Example:
1000x1000 image
1M hidden units
Filter size: 10x10
100M parameters

Local Connectivity

LOCALLY CONNECTED NEURAL NET



Example:
1000x1000 image
1M hidden units
Filter size: 10x10
100M parameters

ConvNets

Convolutional Neural Networks
are just Neural Networks BUT:

1. Local connectivity

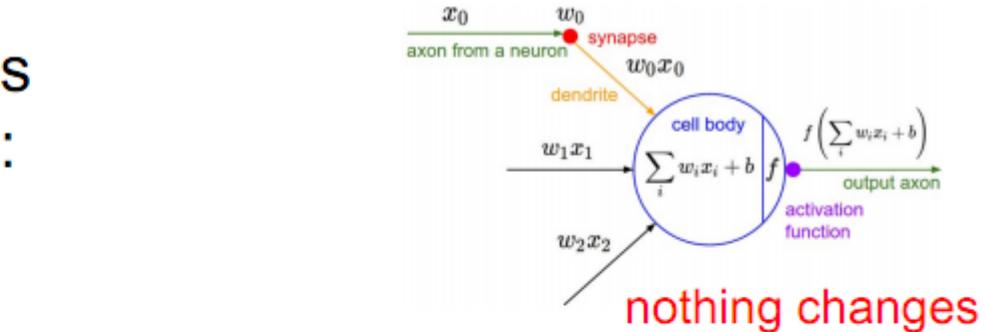
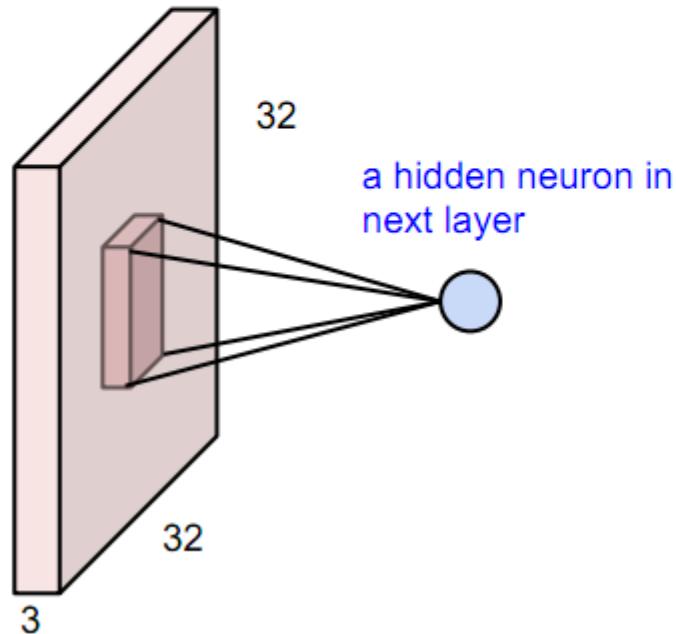


image: 32x32x3 volume

before: full connectivity: 32x32x3 weights

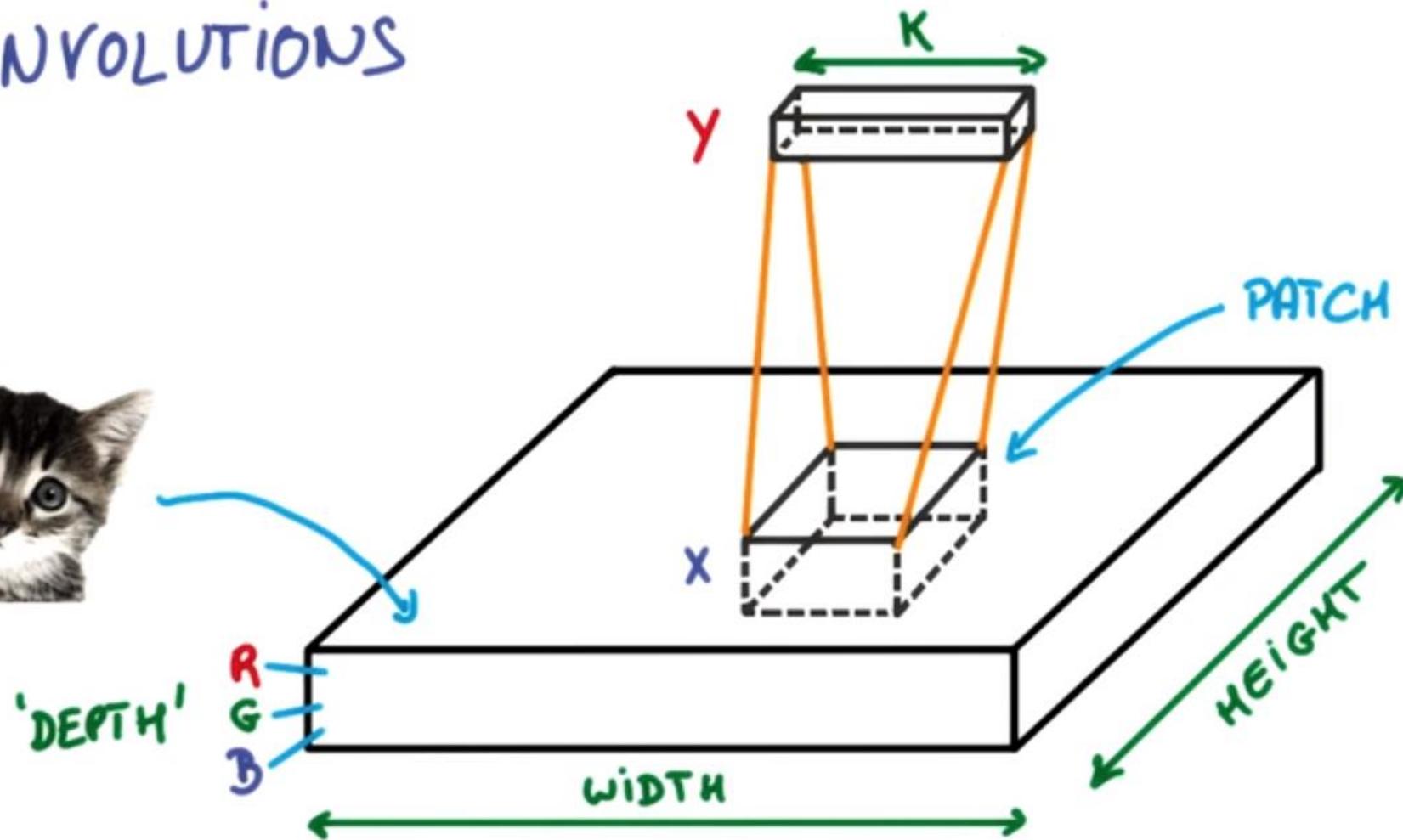
now: one neuron will connect to, e.g. 5x5x3 chunk and only have 5x5x3 weights.

note that connectivity is:

- local in space (5x5 inside 32x32)
- but full in depth (all 3 depth channels)

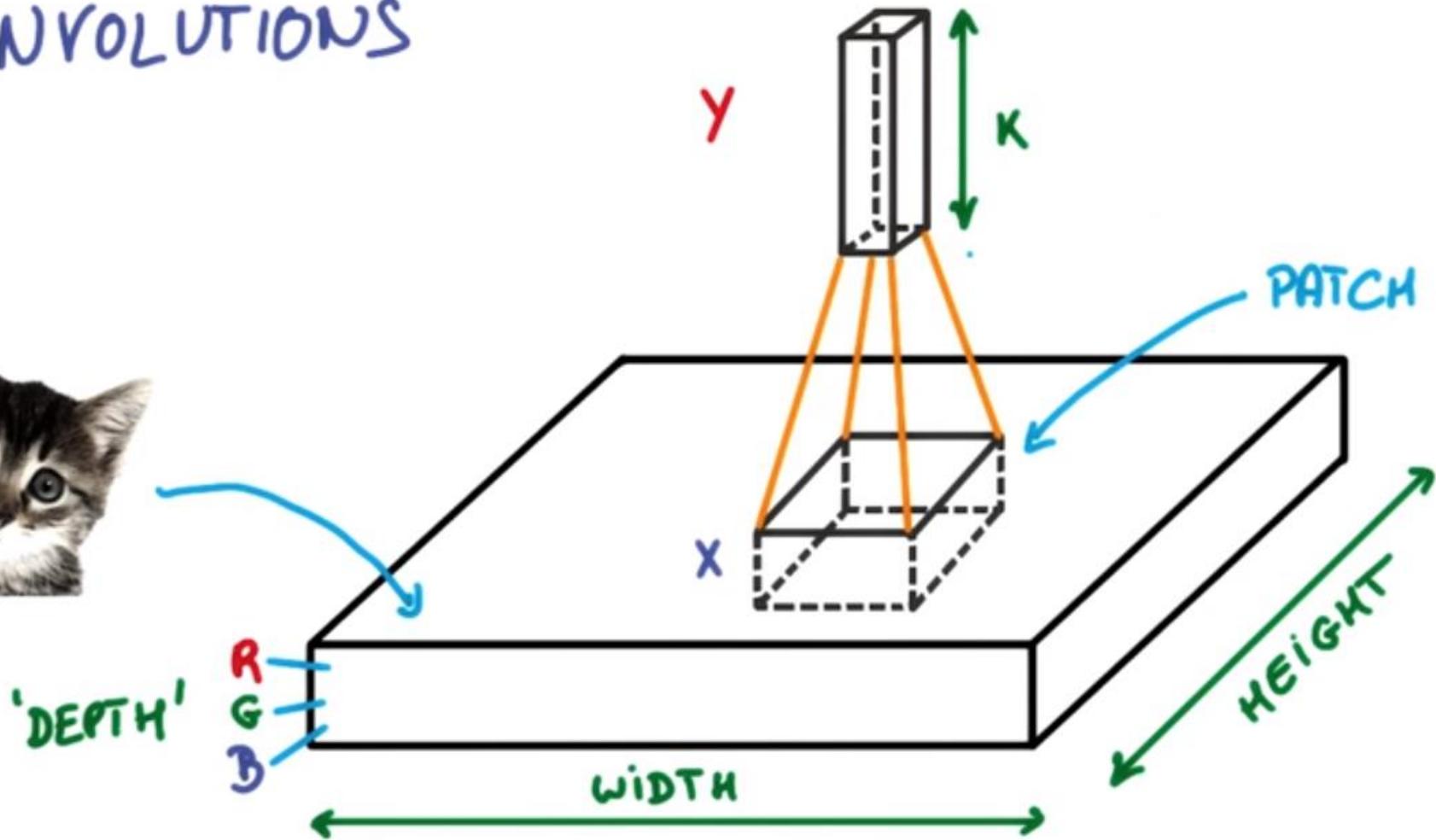
ConvNets

CONVOLUTIONS



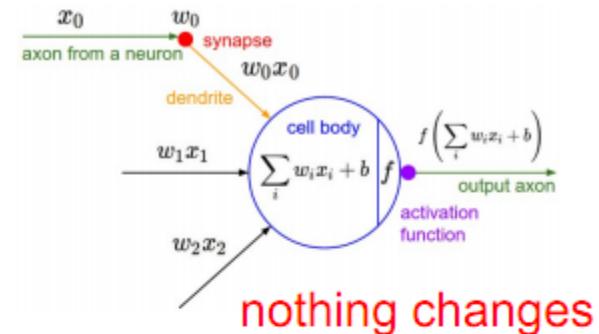
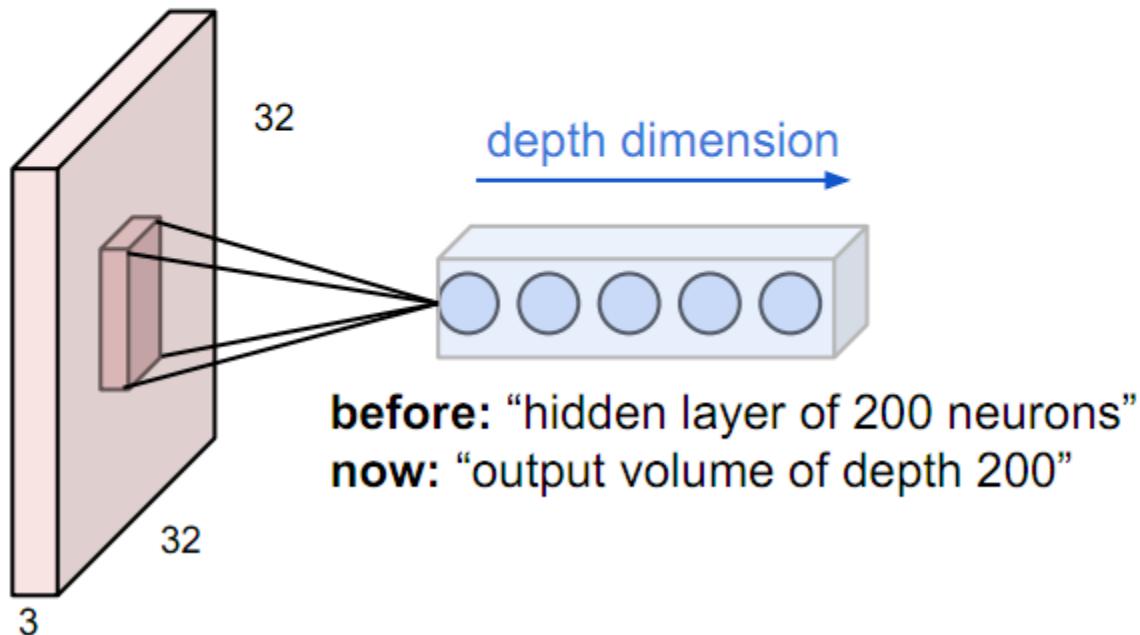
ConvNets

CONVOLUTIONS



ConvNets

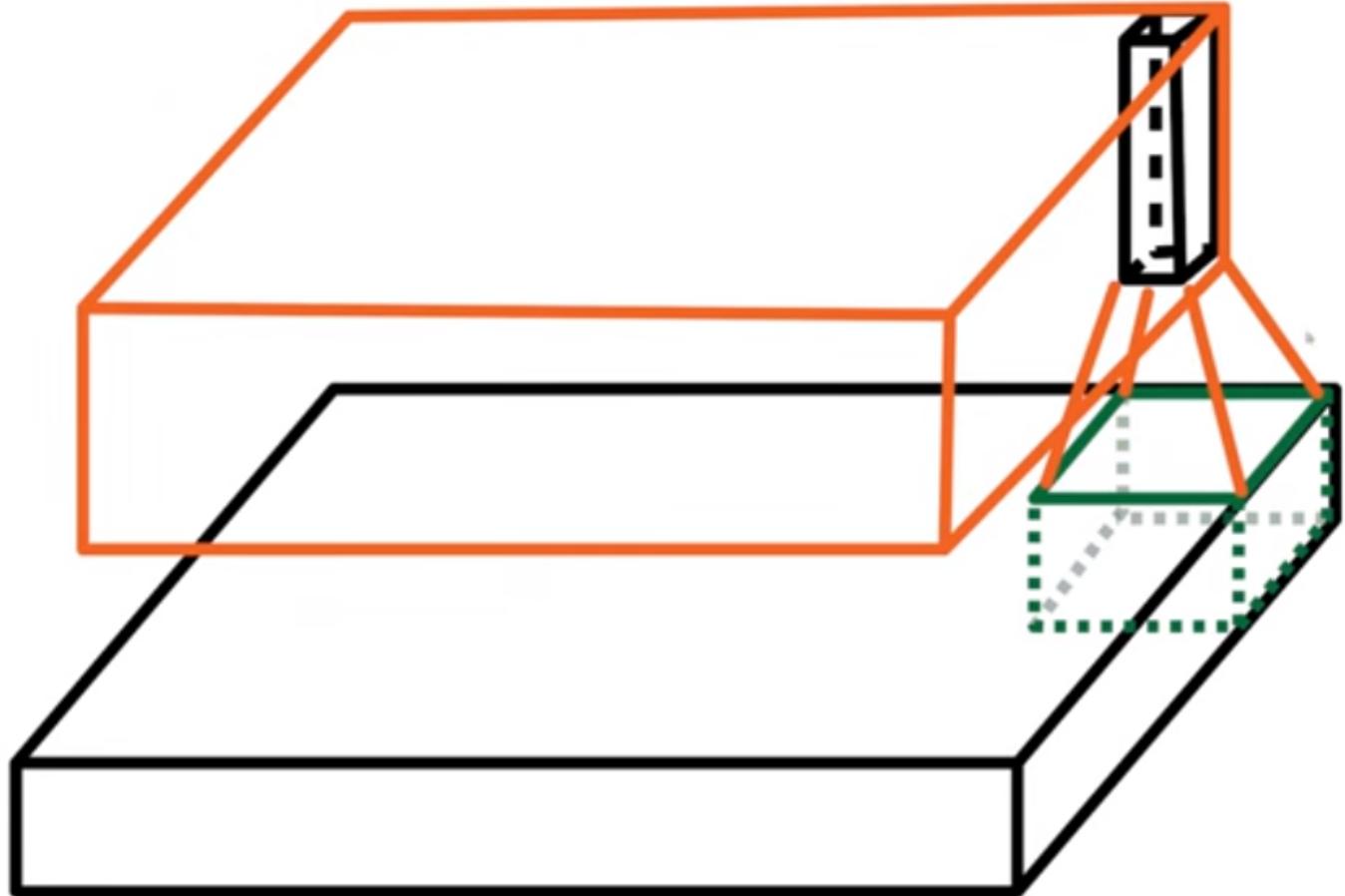
Convolutional Neural Networks
are just Neural Networks BUT:
1. Local connectivity



Multiple neurons all looking at the same region of the input volume, stacked along depth.

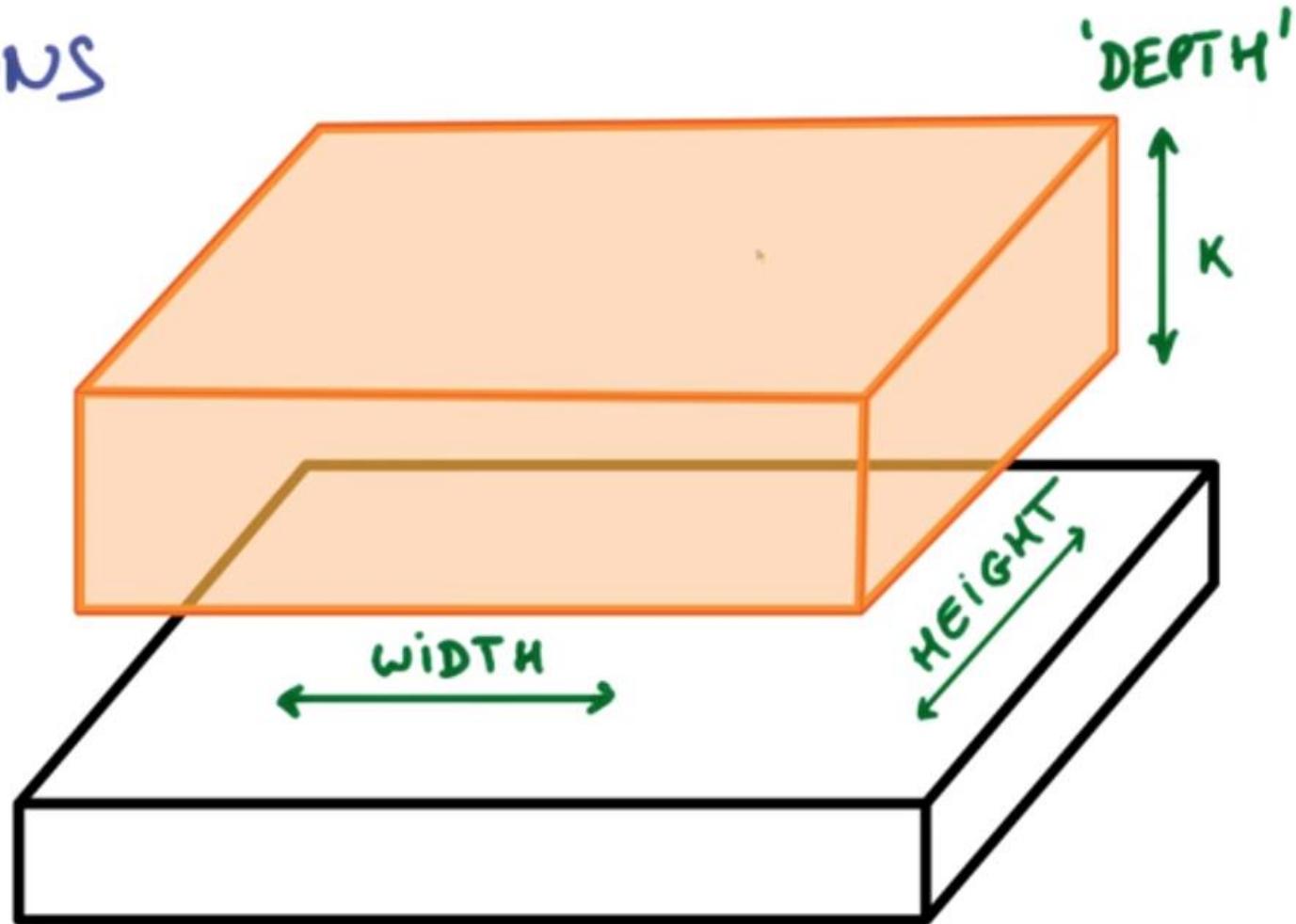
ConvNets

CONVOLUTIONS



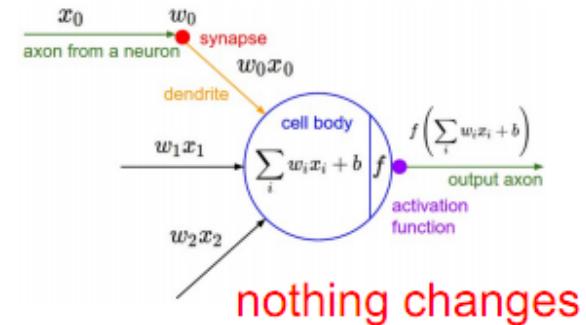
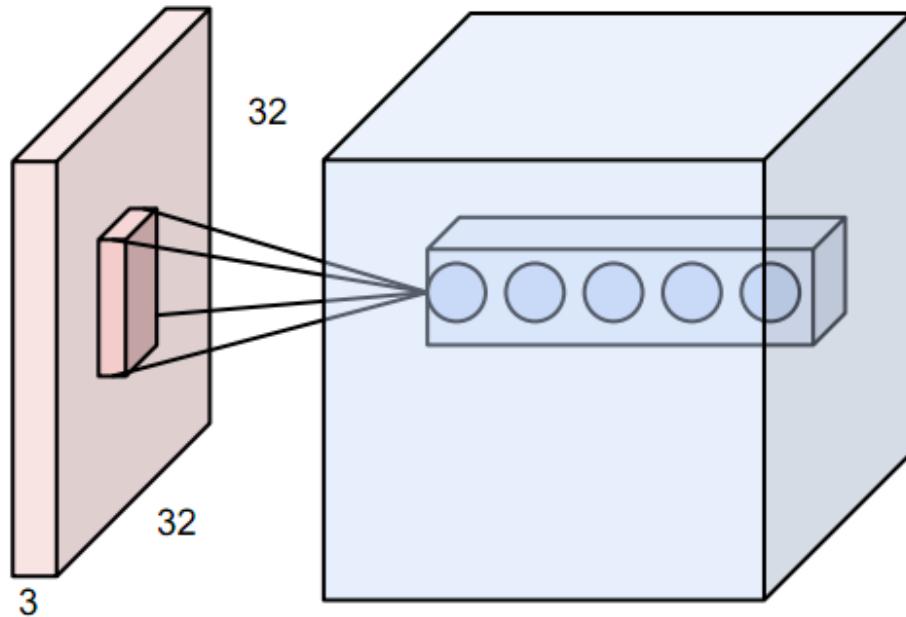
ConvNets

CONVOLUTIONS



ConvNets

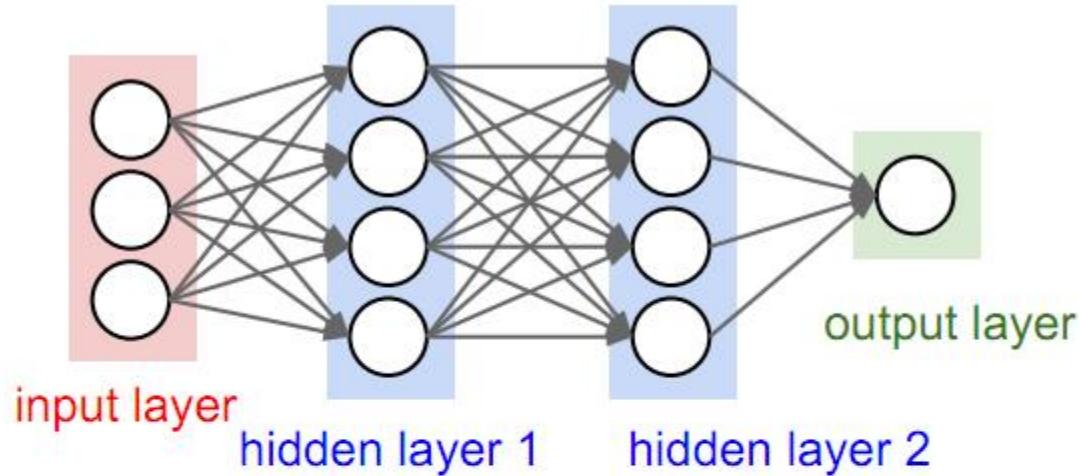
Convolutional Neural Networks
are just Neural Networks BUT:
1. Local connectivity



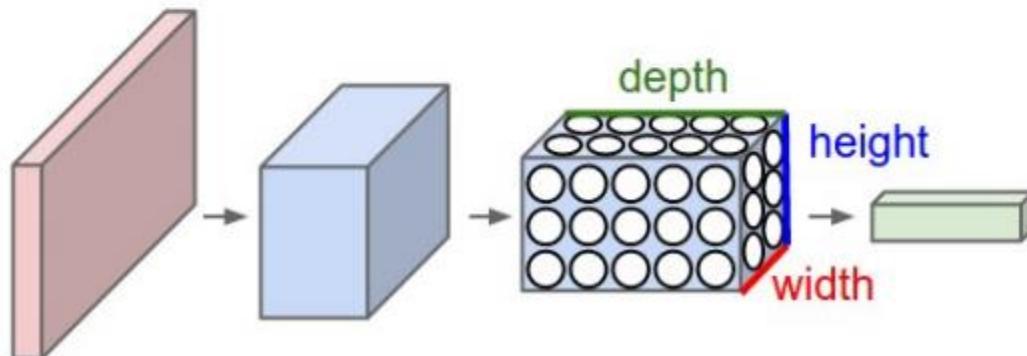
These form a single
[1 x 1 x depth]
“depth column” in the
output volume

ConvNets

before:

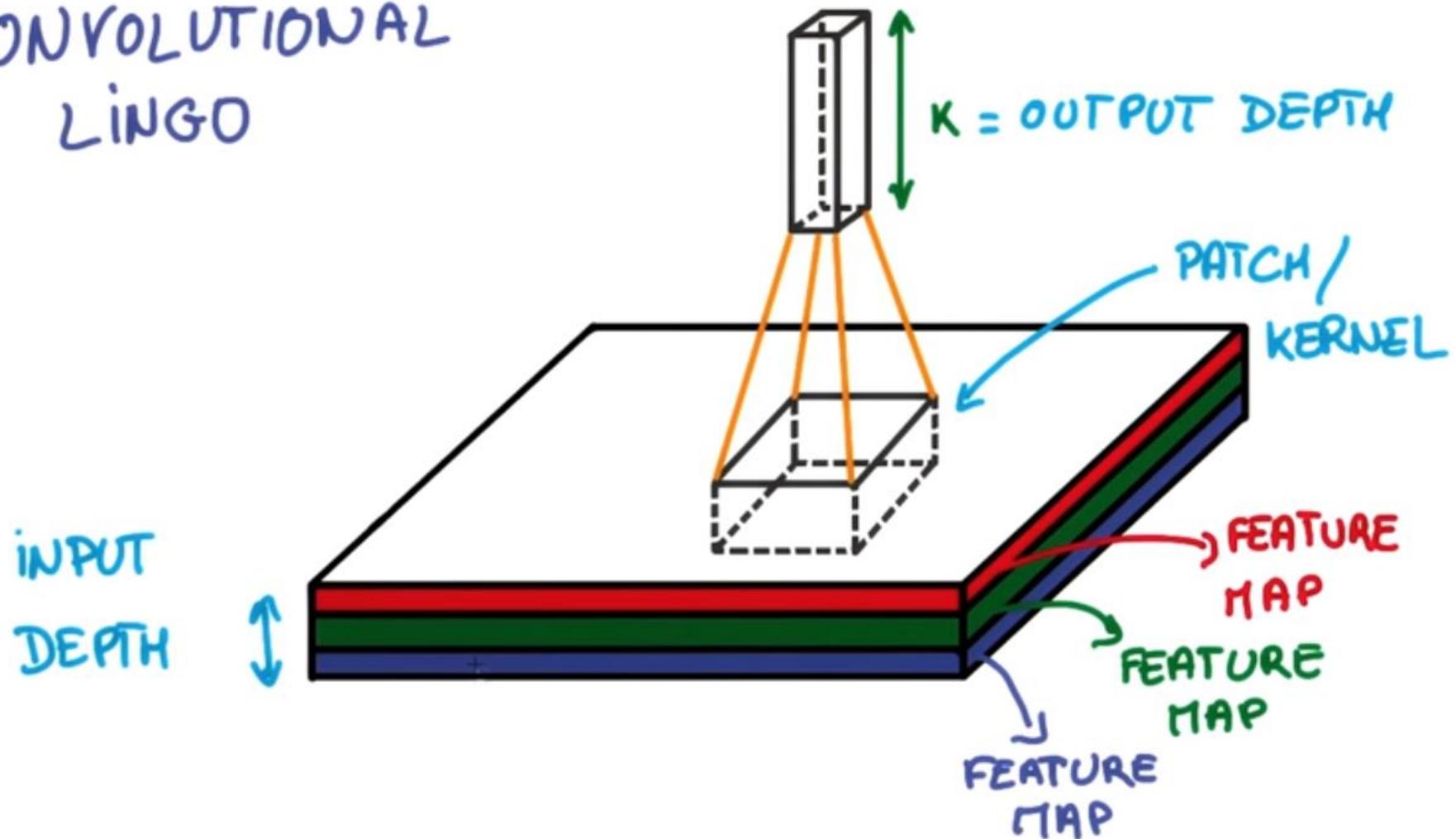


now:



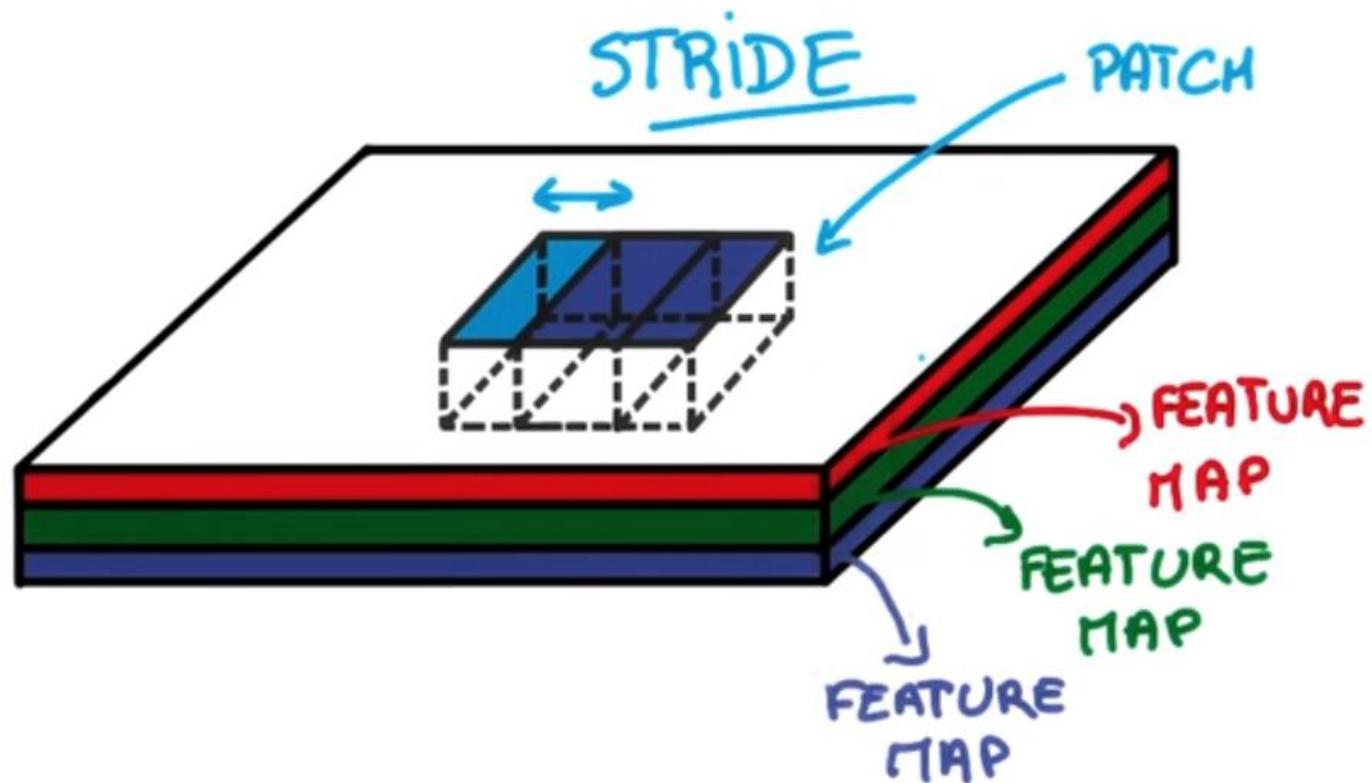
Feature Map

CONVOLUTIONAL
LINGO



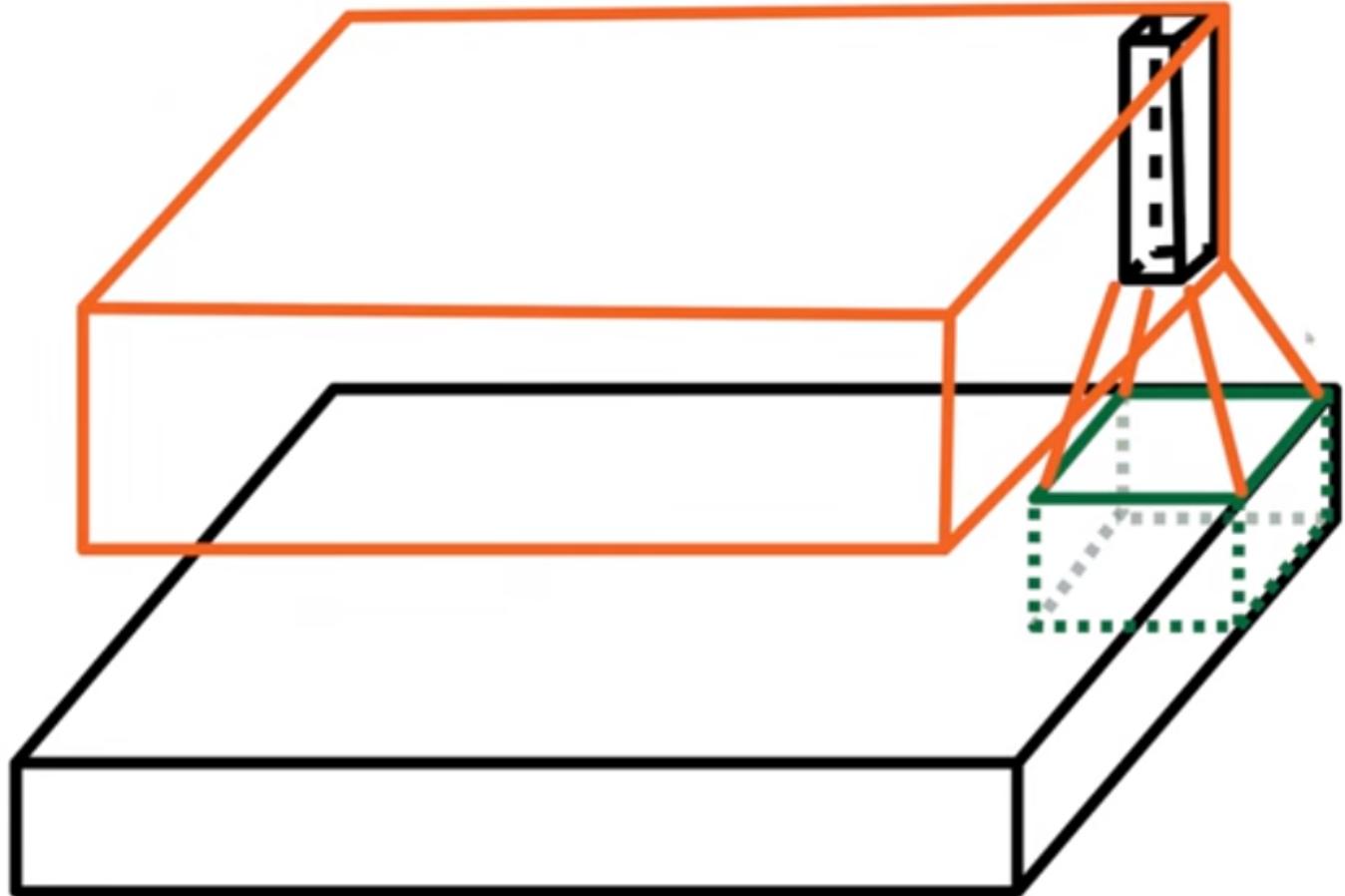
Strike

CONVOLUTIONAL LINGO



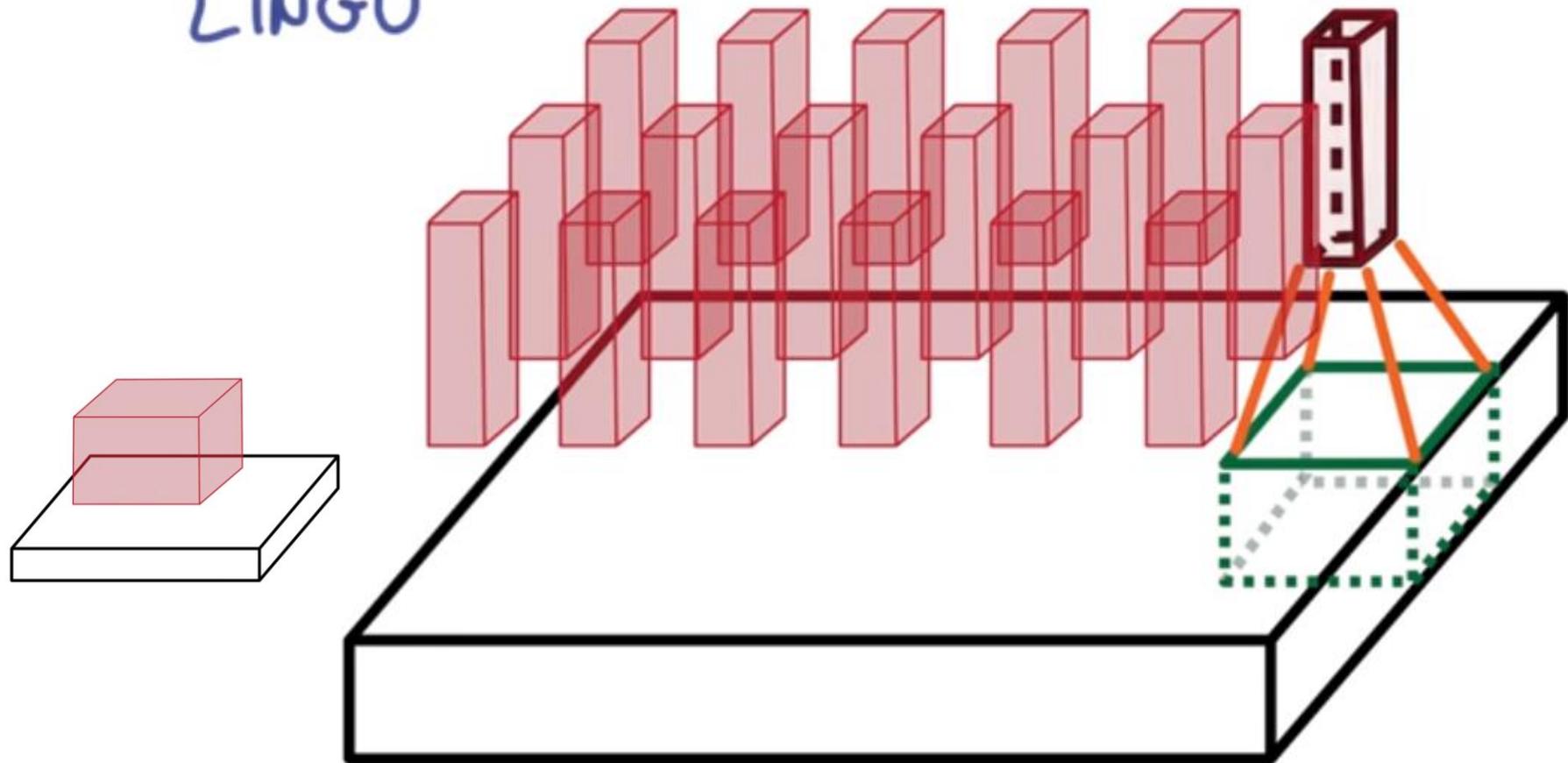
Strike

CONVOLUTIONS



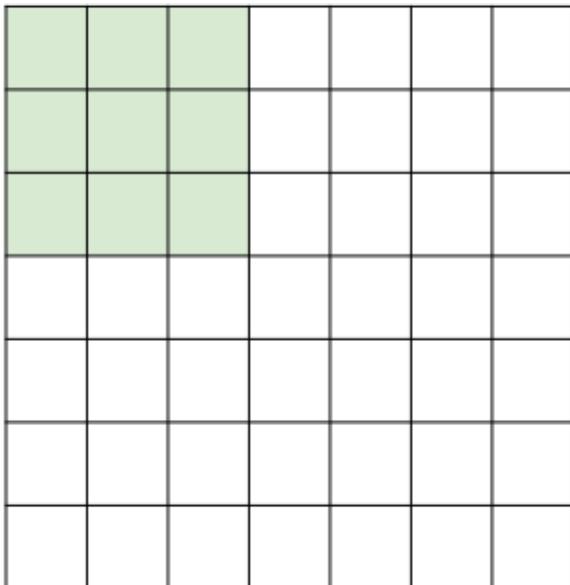
Strike

CONVOLUTIONAL LINGO



Strike

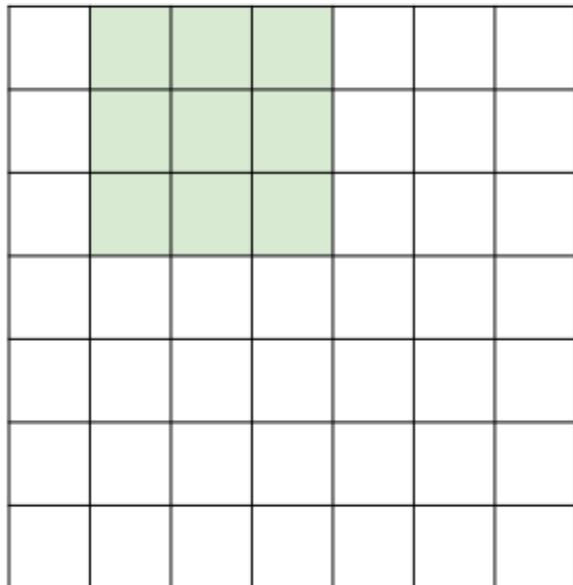
Replicate this column of hidden neurons across space, with some **stride**.



7x7 input
assume 3x3 connectivity, stride 1

Strike

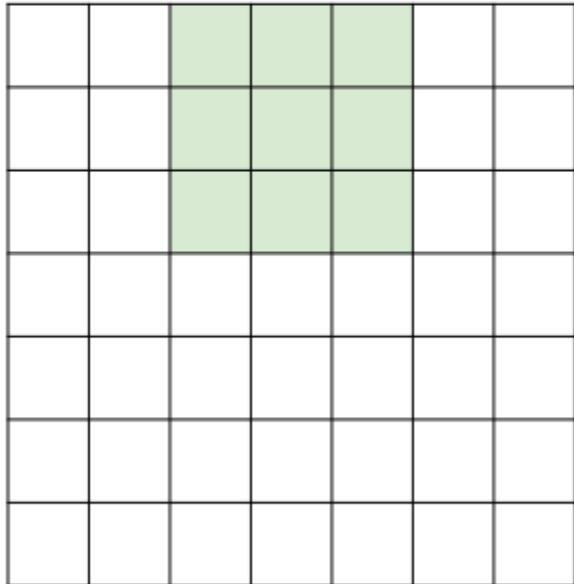
Replicate this column of hidden neurons across space, with some **stride**.



7x7 input
assume 3x3 connectivity, stride 1

Strike

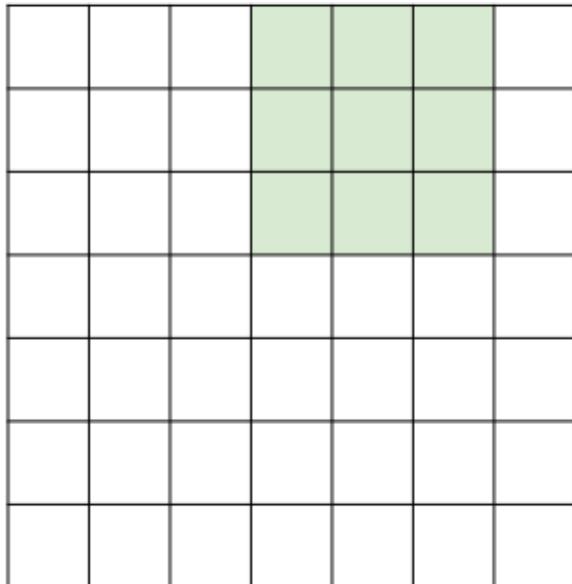
Replicate this column of hidden neurons across space, with some **stride**.



7x7 input
assume 3x3 connectivity, stride 1

Strike

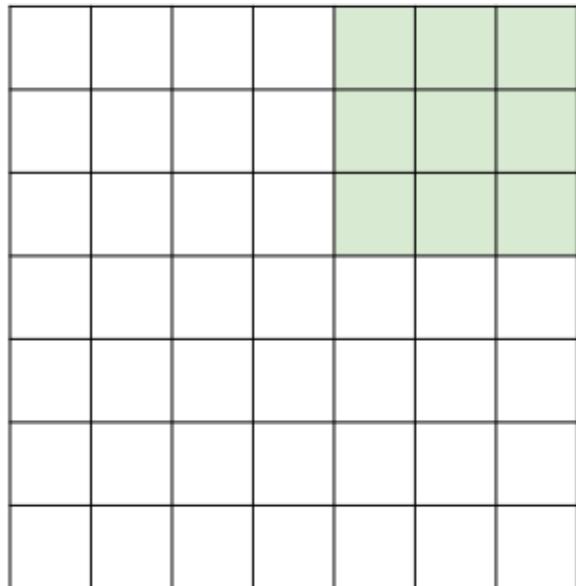
Replicate this column of hidden neurons across space, with some **stride**.



7x7 input
assume 3x3 connectivity, stride 1

Strike

Replicate this column of hidden neurons across space, with some **stride**.



7x7 input
assume 3x3 connectivity, stride 1
=> **5x5 output**

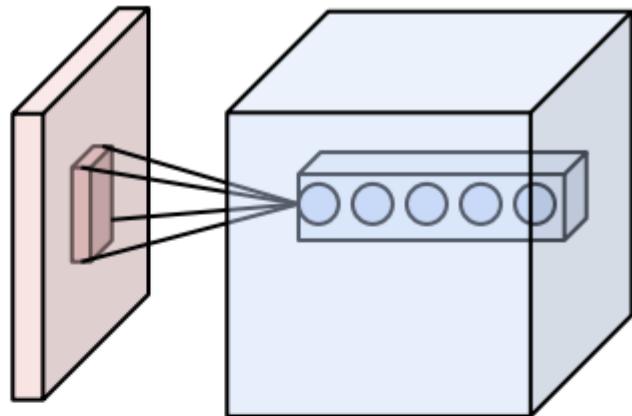
Strike

Examples time:

Input volume: **32x32x3**

Receptive fields: **5x5, stride 1**

Number of neurons: **5**



Output volume: $(32 - 5) / 1 + 1 = 28$, so: **28x28x5**

How many weights for each of the 28x28x5 neurons?

Strike

In practice: Common to zero pad the border

(in each channel)

0	0	0	0	0	0		
0							
0							
0							
0							

e.g. input 7x7

neuron with receptive field 3x3, stride 1
pad with 1 pixel border => what is the output?

7x7 => preserved size!

in general, common to see stride 1, size F, and
zero-padding with $(F-1)/2$.
(Will preserve input size spatially)

Strike

In practice: Common to zero pad the border

(in each channel)

0	0	0	0	0	0		
0							
0							
0							
0							

e.g. input 7x7

neuron with receptive field 3x3, stride 1
pad with 1 pixel border => what is the output?

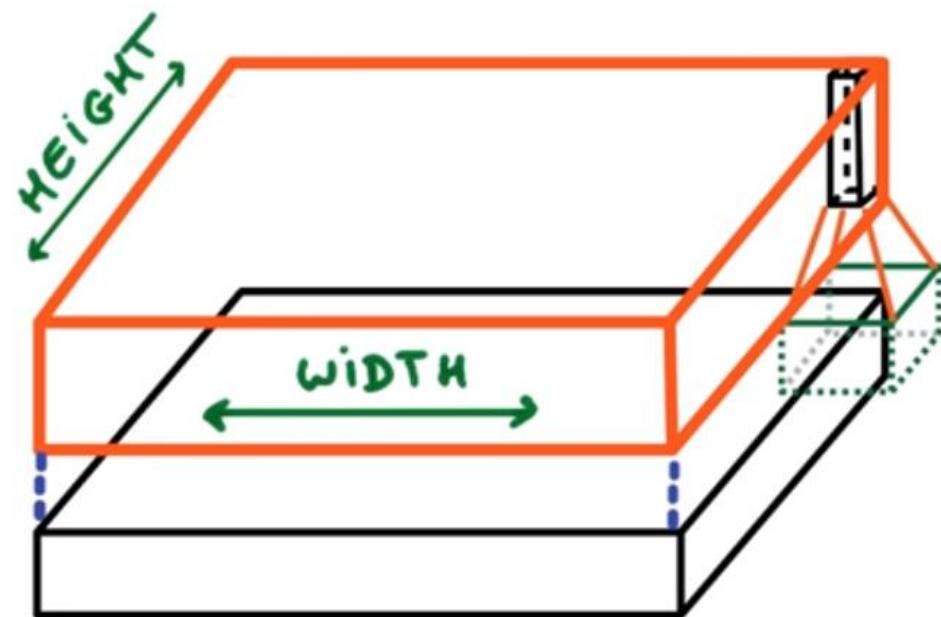
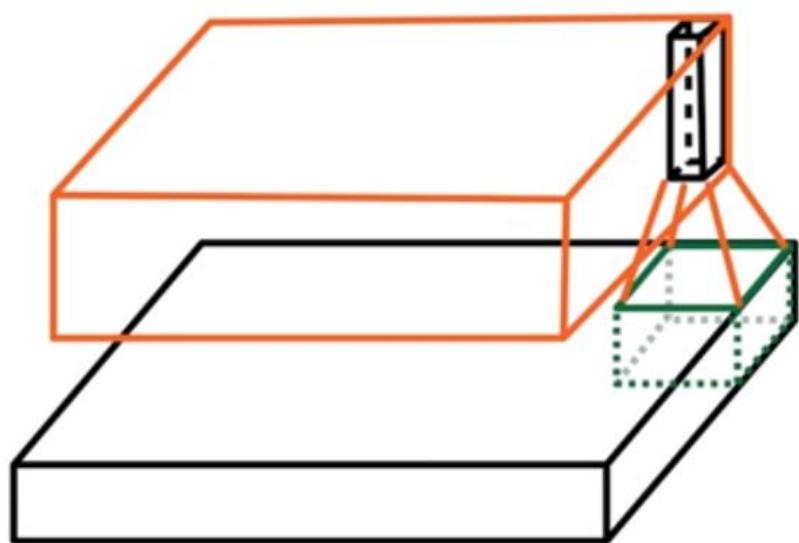
7x7 => preserved size!

in general, common to see stride 1, size F, and
zero-padding with $(F-1)/2$.
(Will preserve input size spatially)

Padding

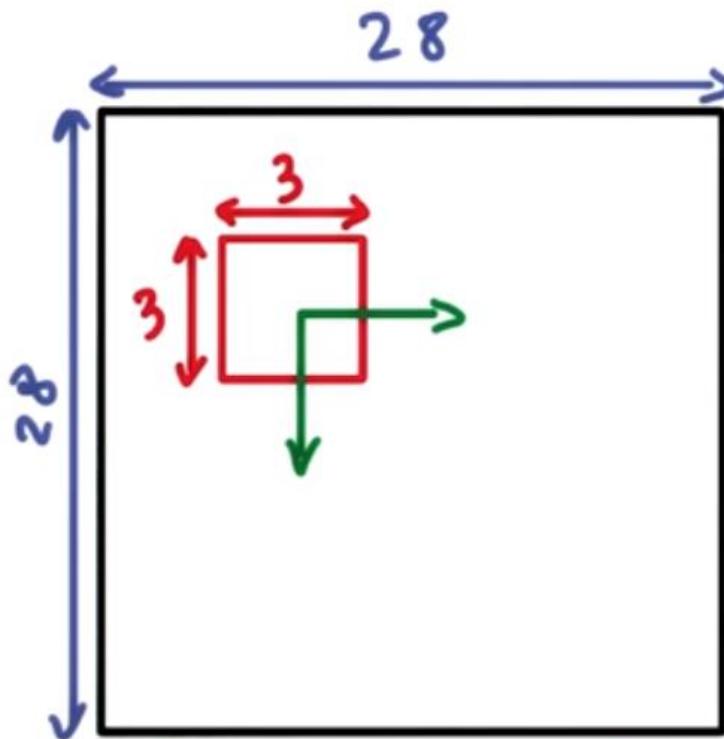
CONVOLUTIONAL
LINGO

'VALID' PADDING
'SAME' PADDING



Quiz

STRIDES, DEPTH & PADDING

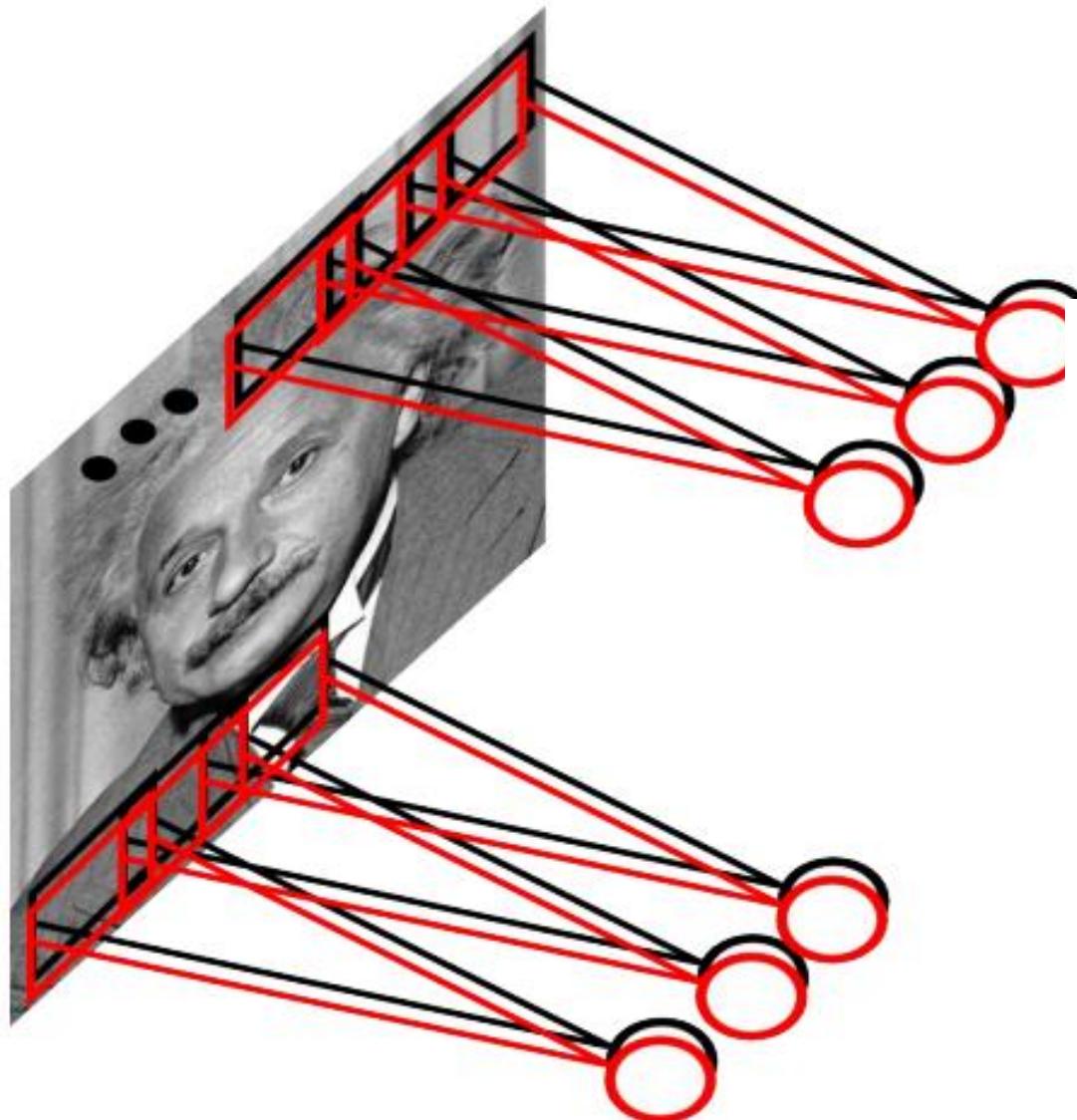


INPUT DEPTH = 3

OUTPUT DEPTH = 8

OUTPUT				
PADDING	STRIDE	WIDTH	HEIGHT	DEPTH
'SAME'		1		
'VALID'		1		
'VALID'	2	2		

ConvNets

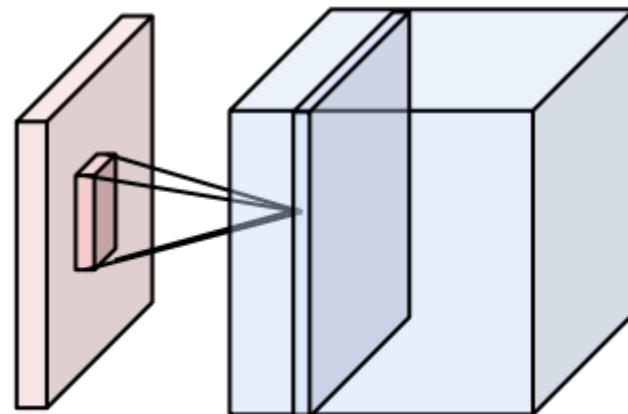


Share
weights

ConvNets

These layers are called **Convolutional Layers**

1. Connect neurons only to local receptive fields
2. Use the same neuron weight parameters for neurons in each “depth slice” (i.e. across spatial positions)

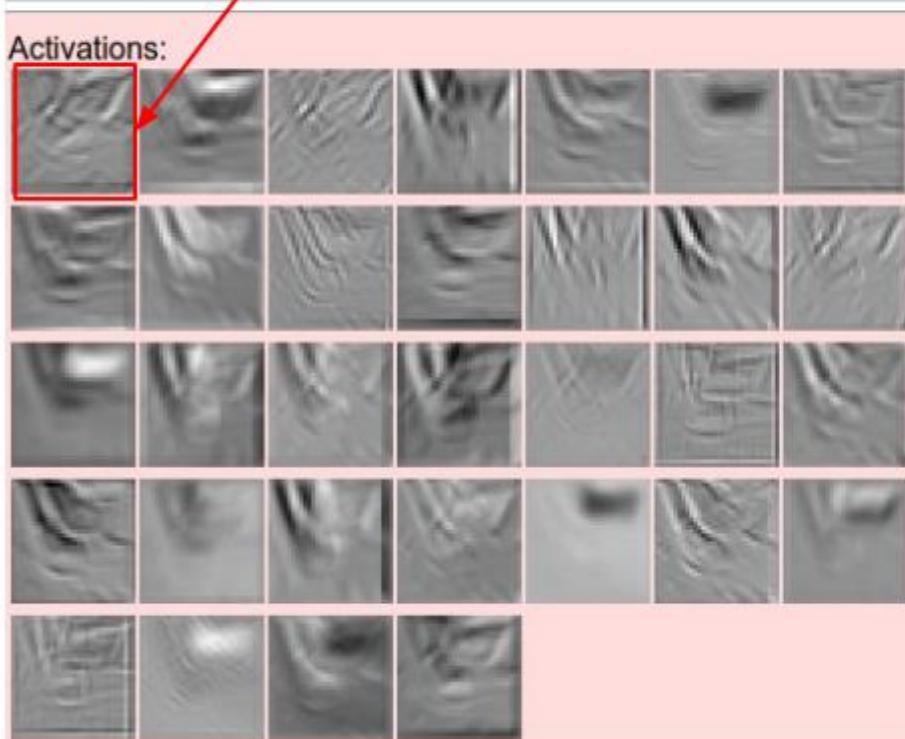


one activation map (a depth slice),
computed with one set of weights

Visualize CNNs



one filter = one depth slice (or activation map)



5x5 filters

Can call the neurons “filters”

We call the layer convolutional because it is related to convolution of two signals (kind of):

$$f[x,y] * g[x,y] = \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} f[n_1, n_2] \cdot g[x - n_1, y - n_2]$$

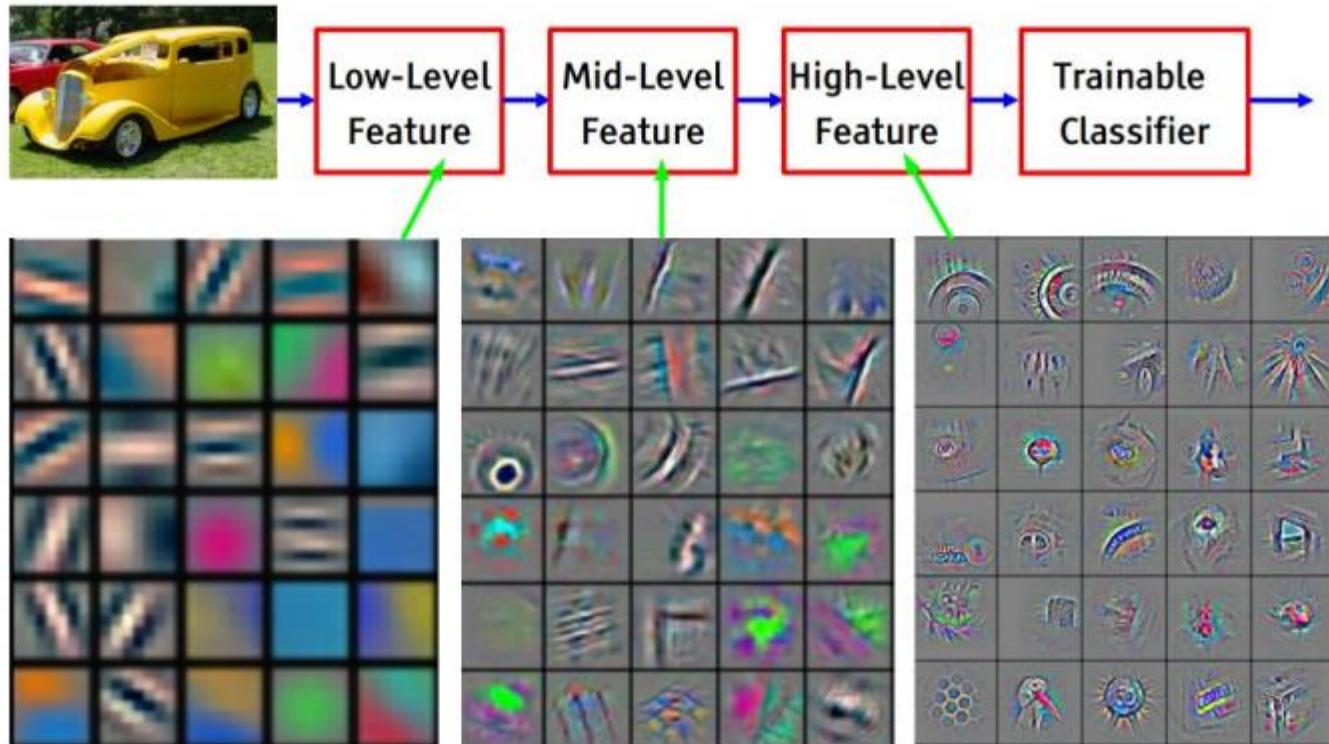


elementwise multiplication and sum
a filter and the signal (image)
 $= \text{np.dot}(w, x) + b$

Visualize CNNs

Fast-forward to today

[From recent Yann LeCun slides]



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

Visualize CNNs

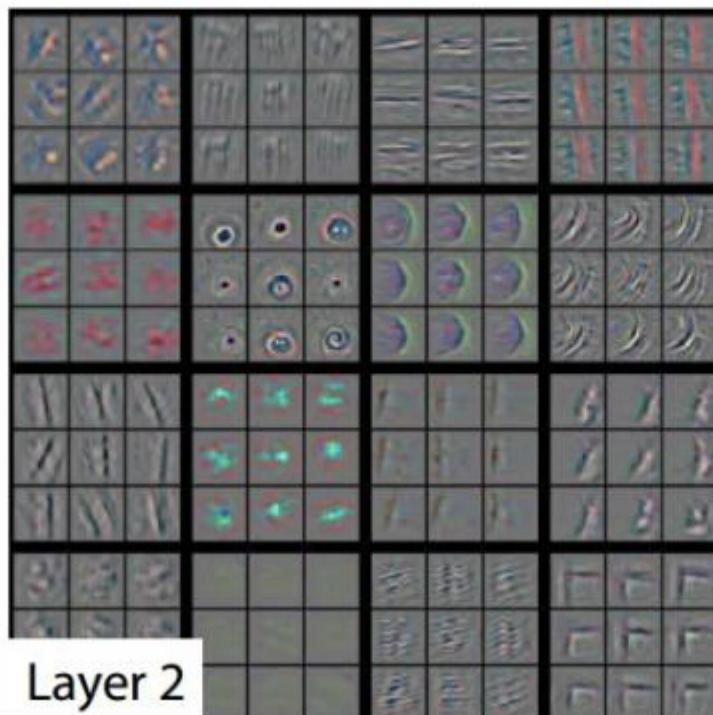
Visualizing and Understanding Convolutional Networks

Zeiler & Fergus, 2013

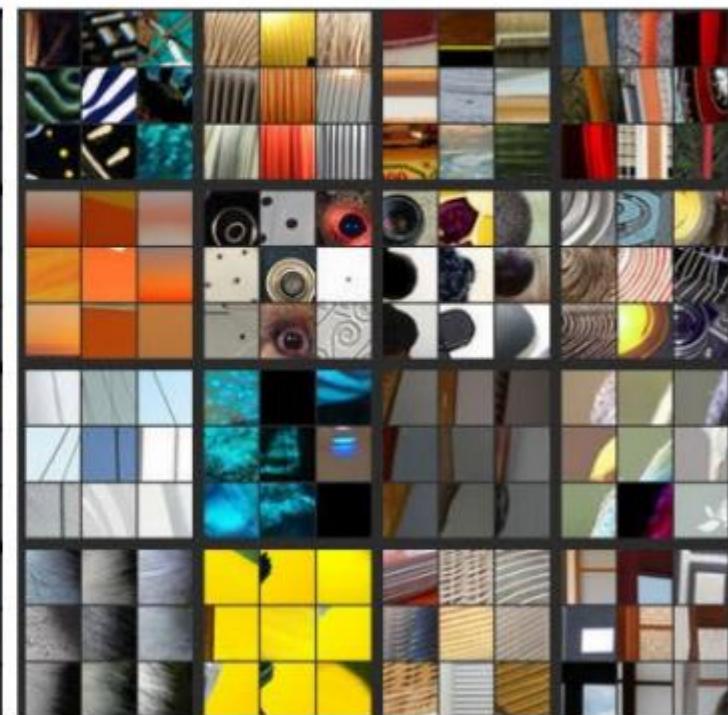
Visualizing arbitrary neurons along the way to the top...



Layer 1

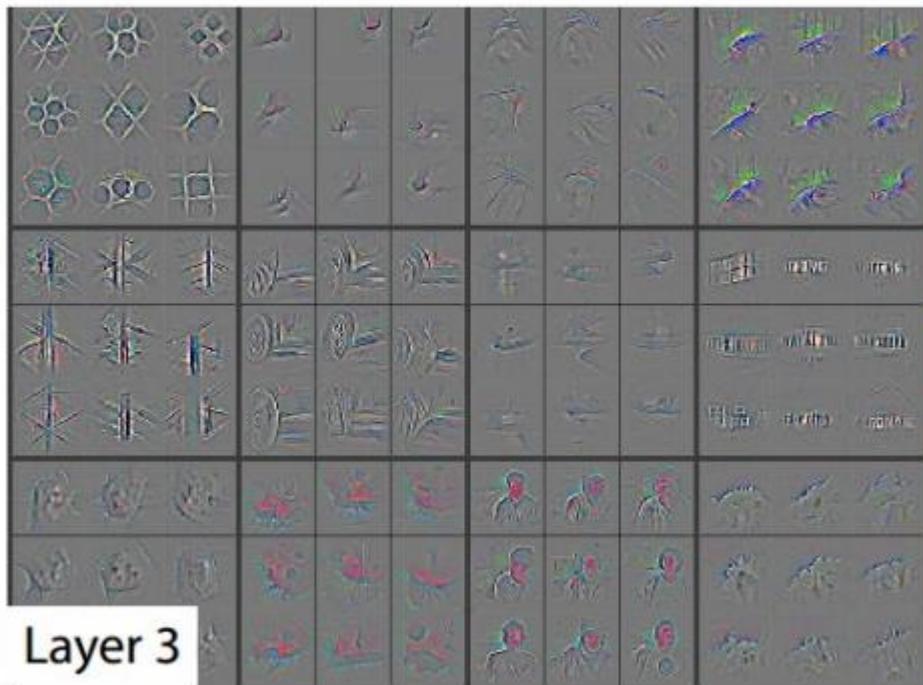


Layer 2



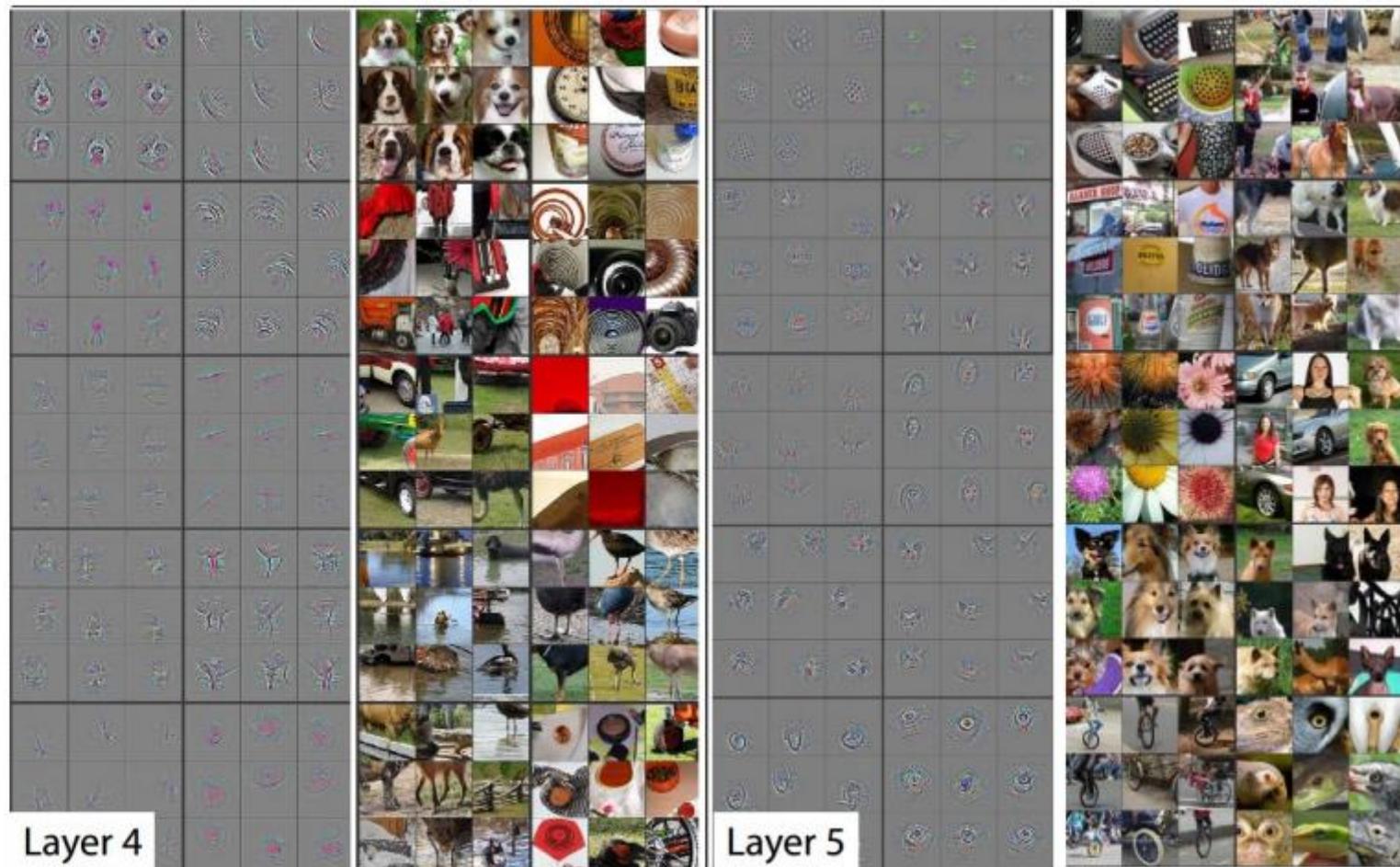
Visualize CNNs

Visualizing arbitrary neurons along the way to the top...

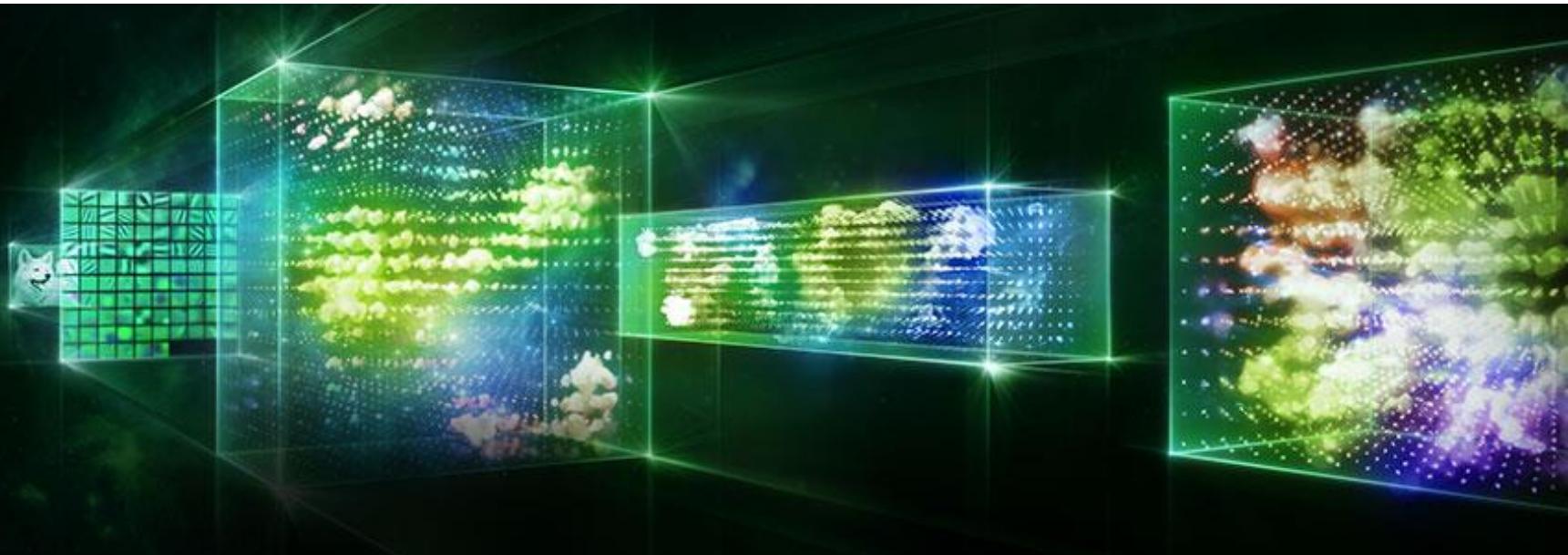


Visualize CNNs

Visualizing arbitrary neurons along the way to the top...



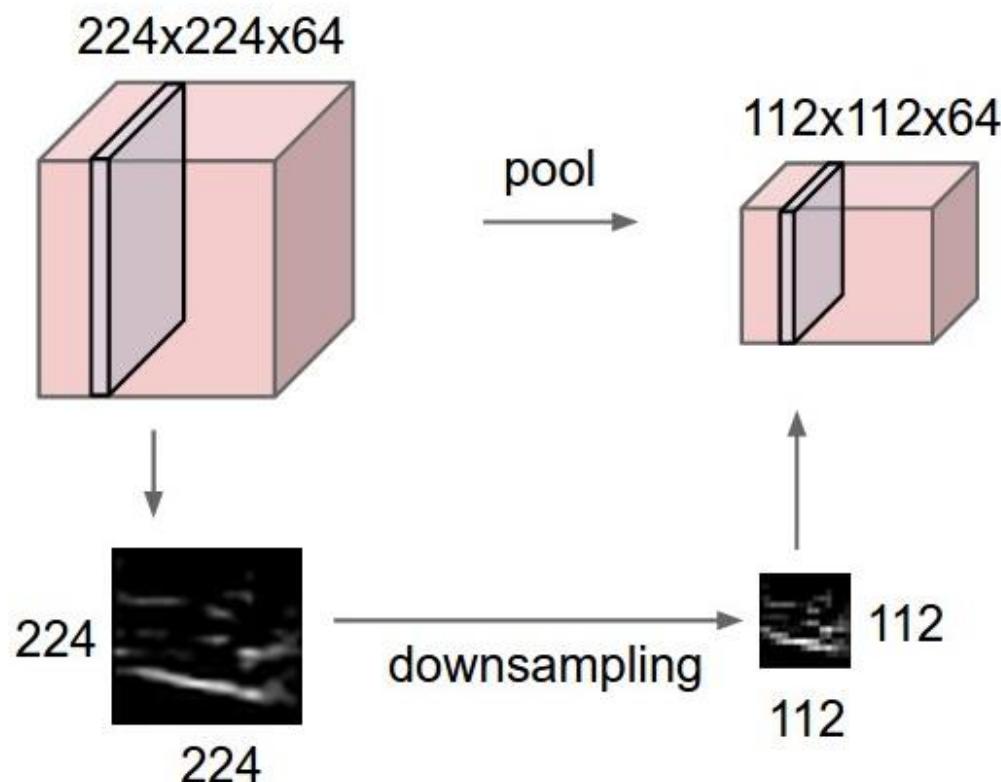
Visualize CNNs



Pooling

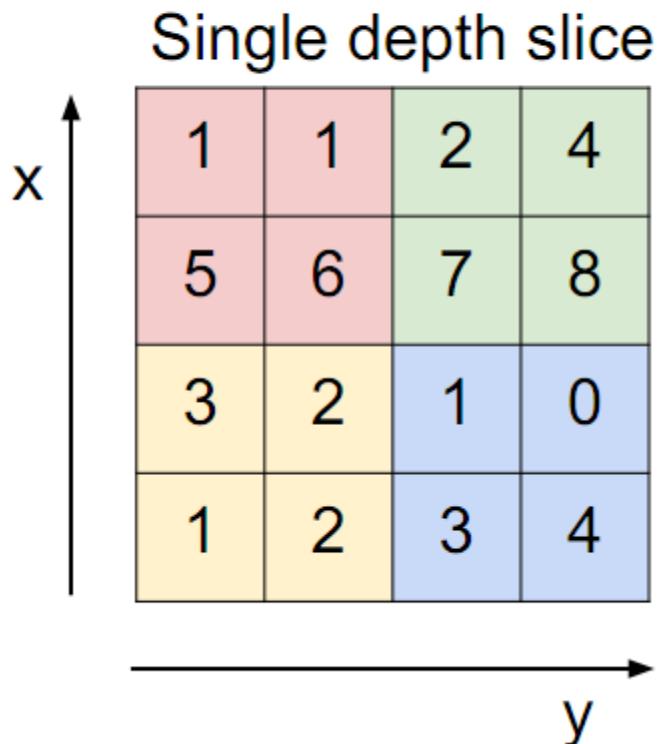
In ConvNet architectures, **Conv** layers are often followed by **Pool** layers

- convenience layer: makes the representations smaller and more manageable without losing too much information. Computes MAX operation (most common)



Pooling

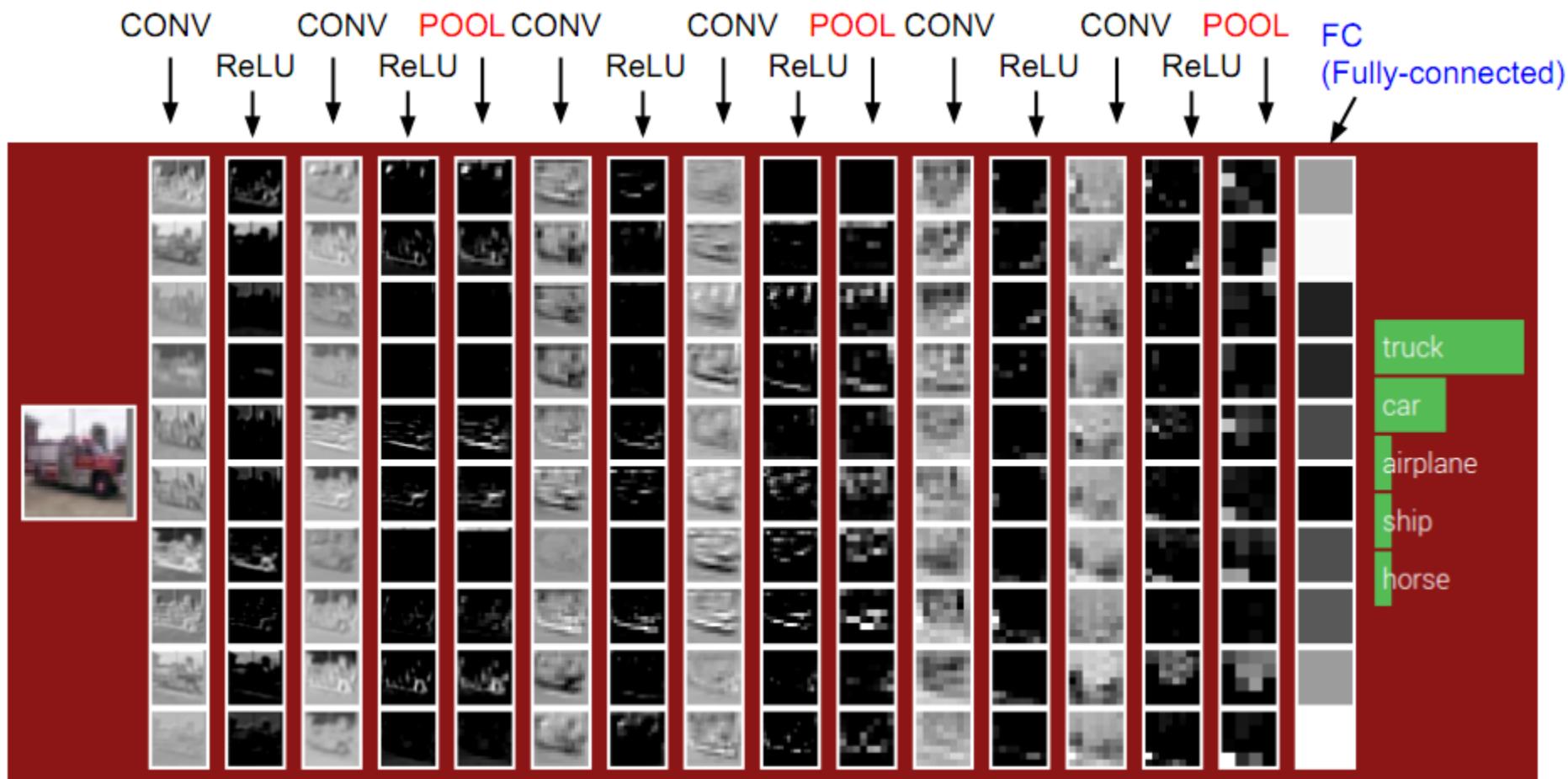
MAX POOLING



max pool with 2x2 filters
and stride 2

6	8
3	4

ConvNets



ConvNets

Modern CNNs:

- use **filter sizes of 3x3** (maybe even 2x2 or 1x1!)
- use **pooling sizes of 2x2** (maybe even less - e.g. fractional pooling!)
- **stride 1**
- **very deep**

INPUT -> [[CONV -> RELU]*N -> POOL?] *M -> [FC -> RELU]*K -> FC

where the * indicates repetition, and the POOL? indicates an optional pooling layer.

N ≥ 0 (and usually N ≤ 3), M ≥ 0 , K ≥ 0 (and usually K < 3).

ConvNets

Case study: VGGNet / OxfordNet
(runner-up winner of ILSVRC 2014)
[Simonyan and Zisserman]

best model

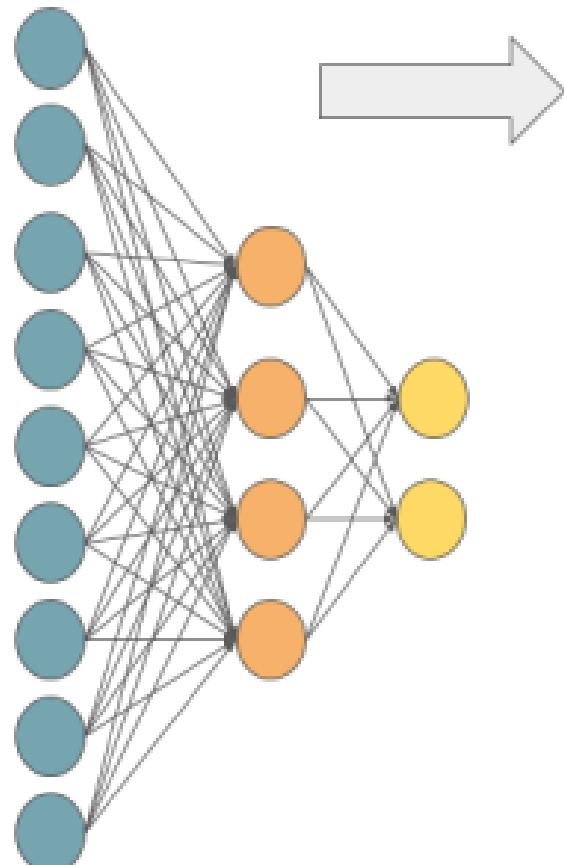
ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224×224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Table 2: Number of parameters (in millions).

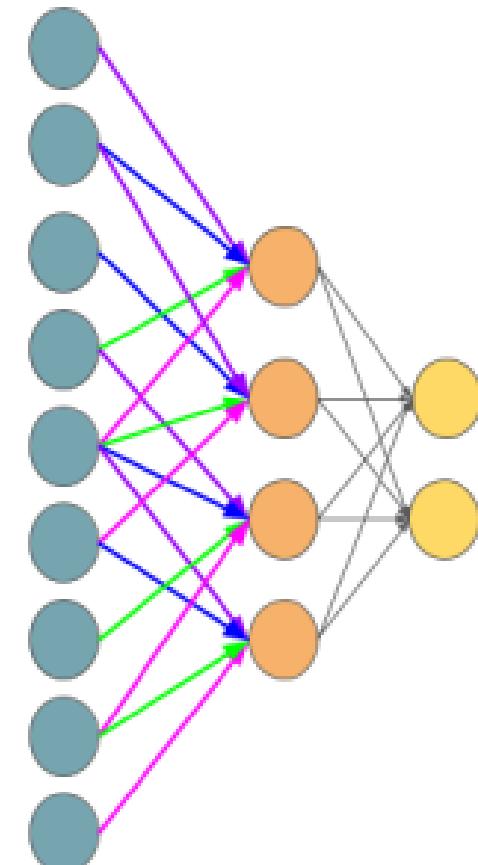
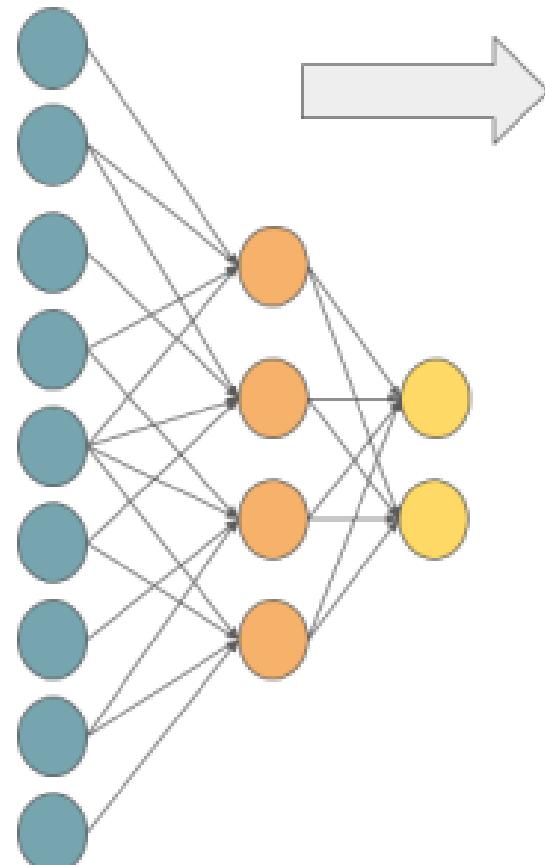
Network	A,A-LRN	B	C	D	E
Number of parameters	133	133	134	138	144

BackPropagation in CNNs

connections cutting



weights sharing



BackPropagation in CNNs

Error signals for each example are computed by upsampling. Upsampling is an operation which backpropagates (distributes) the error signals over the aggregate function g using its derivatives $g'_n = \partial g / \partial x_{(n-1)m+1:nm}$. g'_n can change depending on pooling region n .

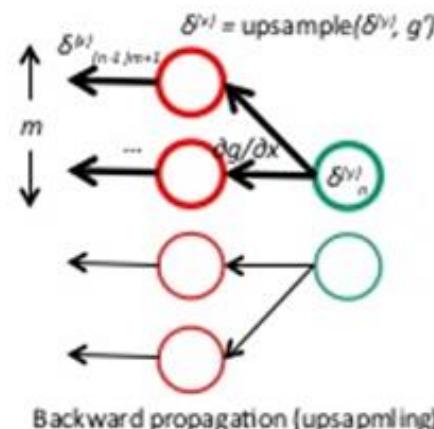
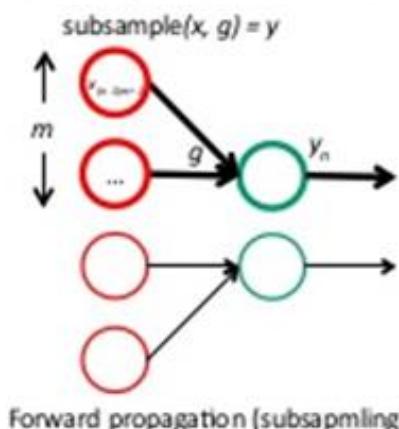
- In max pooling, the unit which was the max at forward propagation receives all the error at backward propagation and the unit is different depending on the region n .

■ Definition: *upsample*

$\text{upsample}(f, g)[n]$ denotes the n -th element of $\text{upsample}(f, g)$.

$$\delta_{(n-1)m+1:nm}^{(i)} = \text{upsample}\left(\delta^{(i)}, g'\right)[n] = \delta_n^{(i)} g'_n = \delta_n^{(i)} \frac{\partial g}{\partial x_{(n-1)m+1:nm}} = \frac{\partial J}{\partial y_n} \frac{\partial y_n}{\partial x_{(n-1)m+1:nm}} = \frac{\partial J}{\partial x_{(n-1)m+1:nm}}$$

$$\delta^{(i)} = \text{upsample}\left(\delta^{(i)}, g'\right) = \left[\delta_{(n-1)m+1:nm}^{(i)} \right]$$



References

1. CNNs for Visual Recognition by Stanford

http://cs231n.stanford.edu/slides/winter1516_lecture7.pdf

2. Deep Learning Course by Udacity

<https://classroom.udacity.com/courses/ud730/lessons/6377263405/concepts/63741833610923#>



TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

Thank you!