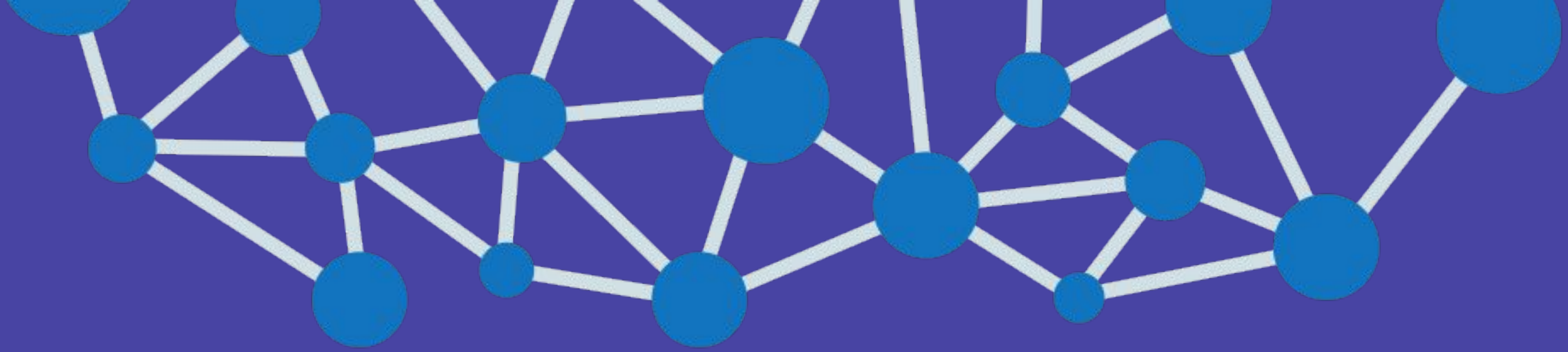


Practical introduction to CNNs

Mariya Hirna, Avenga

Let’s meet

Junior Data Scientist @ Avenga

Ukrainian Catholic University student

GDG Lviv Lead

What about you?

Python, GPUs, Colab & Jupyter

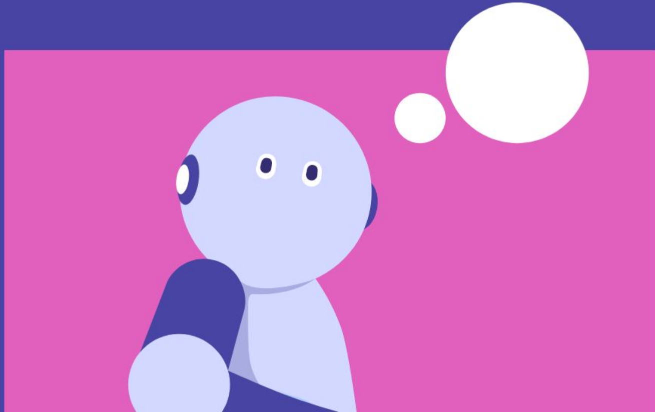
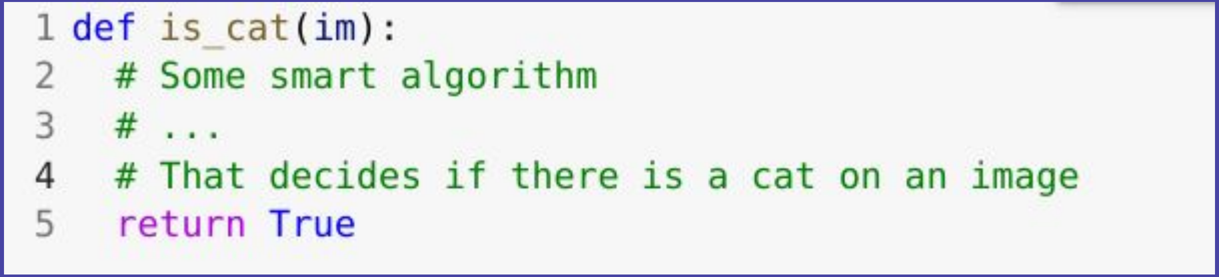


Image classification?



```python
def is_cat(im):
  # Some smart algorithm
  # ...
  # That decides if there is a cat on an image
  return True
```

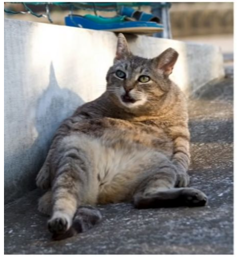
Dataset

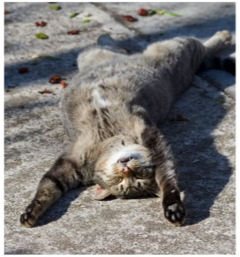
This image is CC0 1.0 public domain

This image is CC0 1.0 public domain

This image is CC0 1.0 public domain

This image is CC0 1.0 public domain


This image by Umberto Salvagnin
is licensed under CC-BY 2.0


This image by Umberto Salvagnin
is licensed under CC-BY 2.0

This image by sare bear is
licensed under CC-BY 2.0

This image is CC0 1.0 public domain

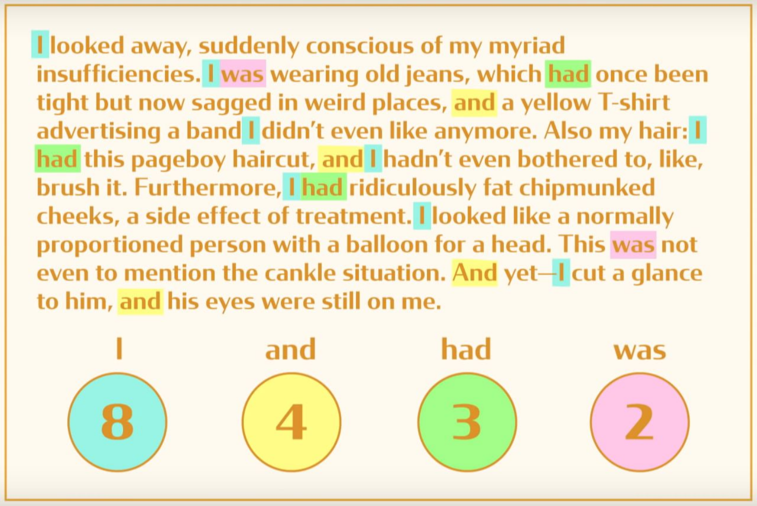
This image is CC0 1.0 public domain

This image is CC0 1.0 public domain

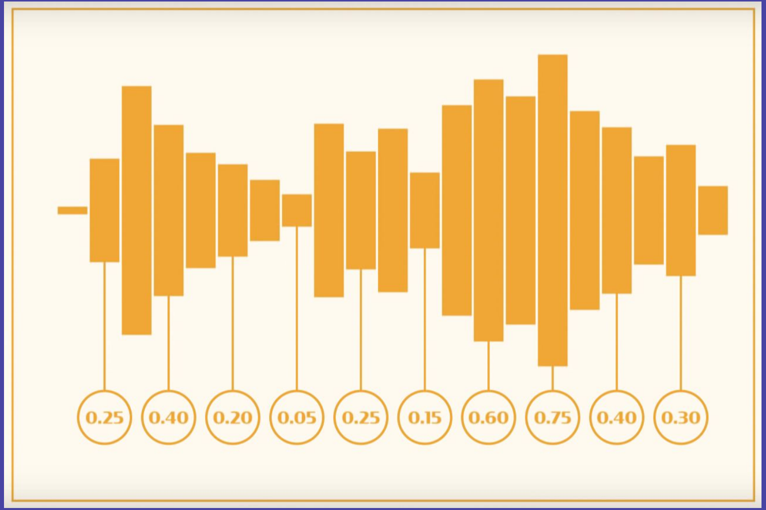
This image is CC0 1.0 public domain

This image by jonsson is licensed
under CC-BY 2.0

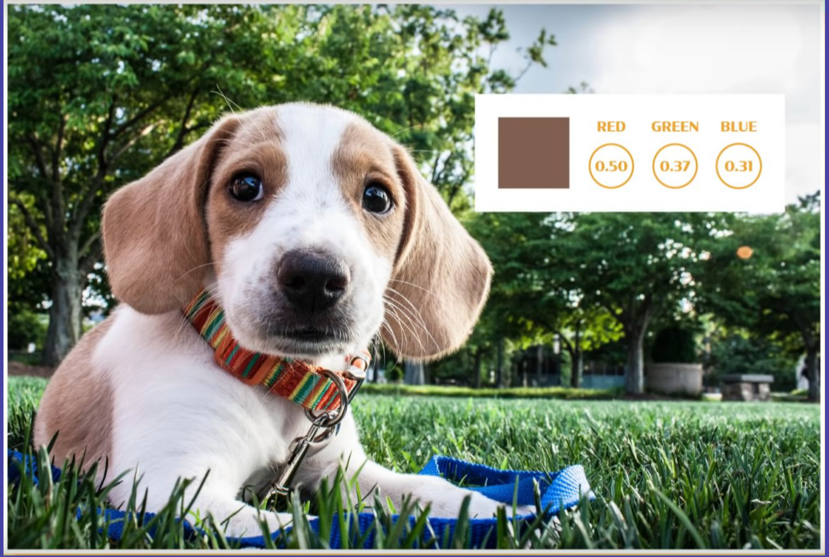
Representing the data

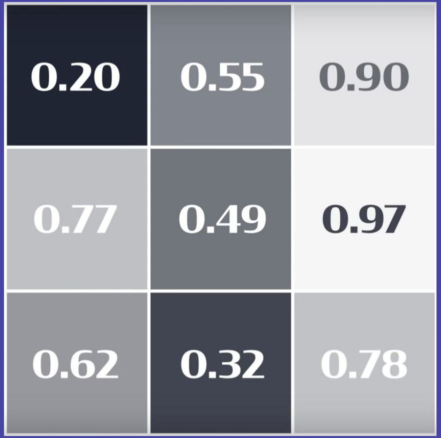


I looked away, suddenly conscious of my myriad insufficiencies. I was wearing old jeans, which had once been tight but now sagged in weird places, and a yellow T-shirt advertising a band I didn't even like anymore. Also my hair: I had this pageboy haircut, and I hadn't even bothered to, like, brush it. Furthermore, I had ridiculously fat chipmunked cheeks, a side effect of treatment. I looked like a normally proportioned person with a balloon for a head. This was not even to mention the cankle situation. And yet—I cut a glance to him, and his eyes were still on me.

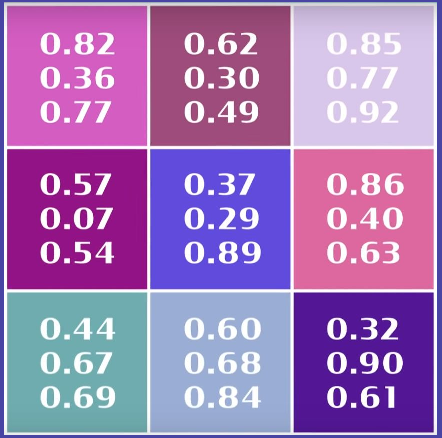


Pixel Data





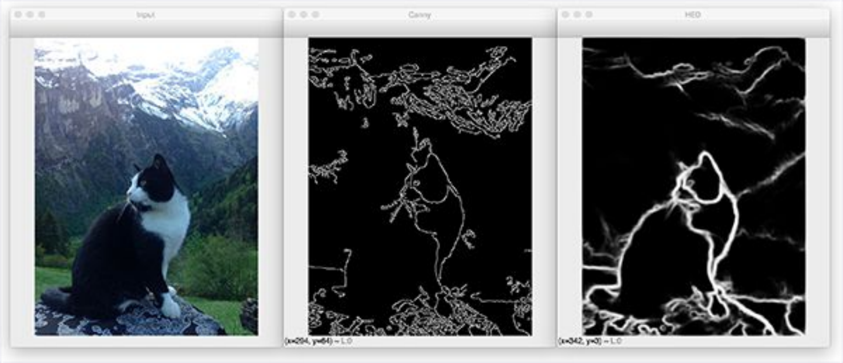
0.20	0.55	0.90
0.77	0.49	0.97
0.62	0.32	0.78



0.82 0.36 0.77	0.62 0.30 0.49	0.85 0.77 0.92
0.57 0.07 0.54	0.37 0.29 0.89	0.86 0.40 0.63
0.44 0.67 0.69	0.60 0.68 0.84	0.32 0.90 0.61

Image classification?

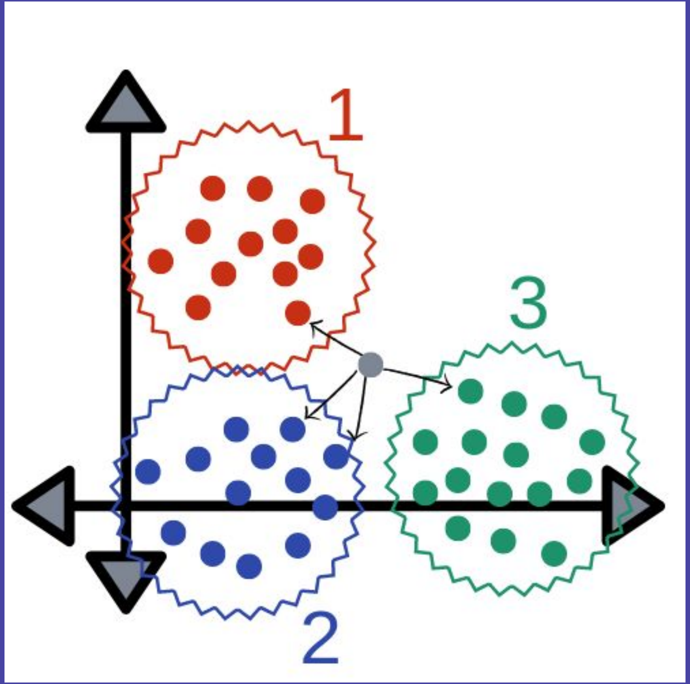
Look for ears, head, colors etc.



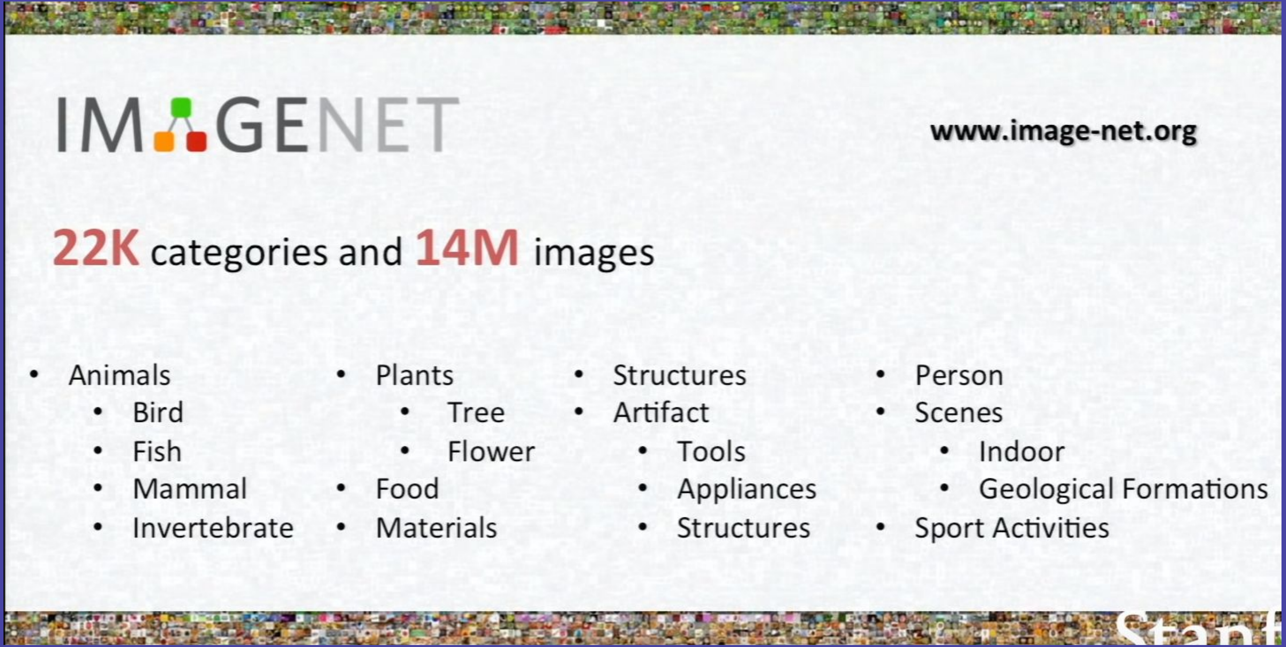
Problem?

Image classification?

K Nearest Neighbours?



ImageNet

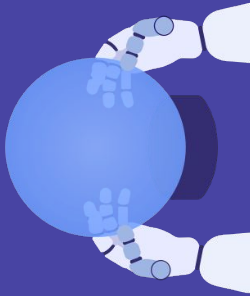


IMAGENET

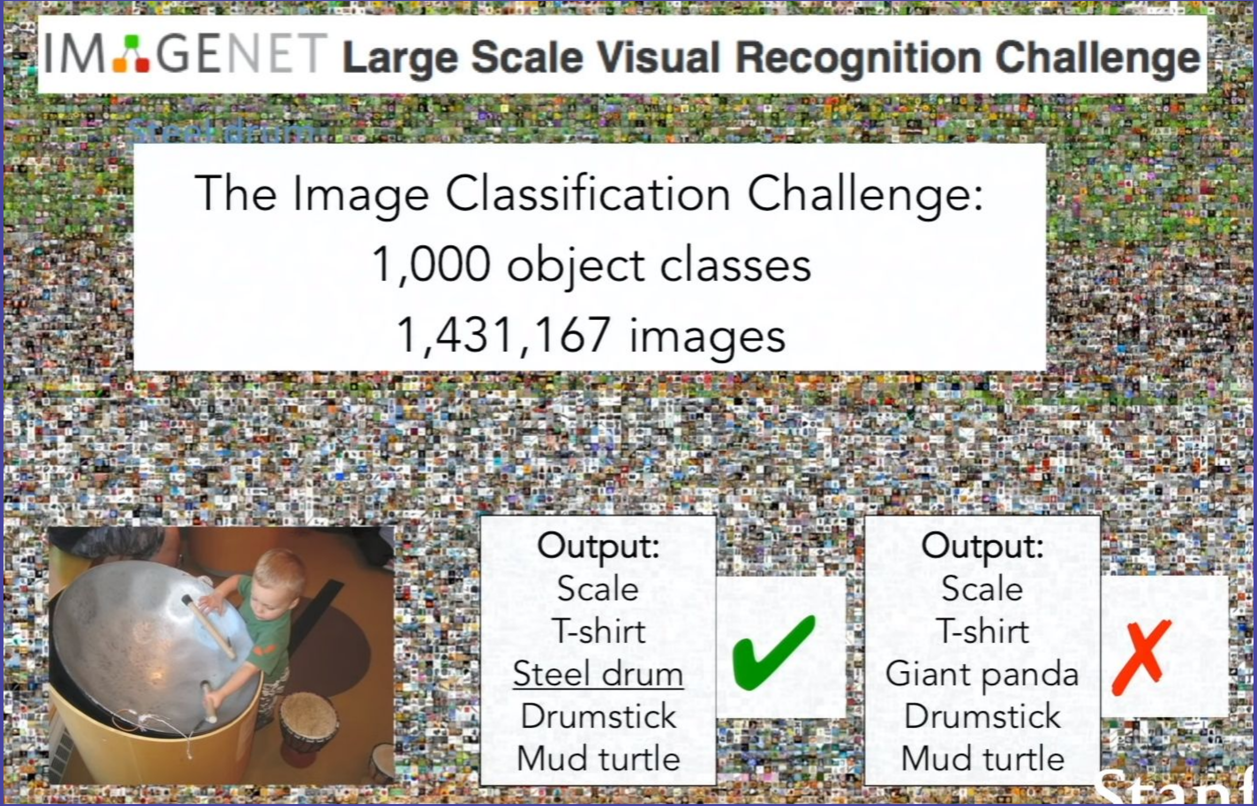
www.image-net.org

22K categories and **14M** images

- Animals
  - Bird
  - Fish
  - Mammal
  - Invertebrate
- Plants
  - Tree
  - Flower
- Food
- Materials
- Structures
- Artifact
  - Tools
  - Appliances
  - Structures
- Person
- Scenes
  - Indoor
  - Geological Formations
- Sport Activities



ImageNet



IMAGENET **Large Scale Visual Recognition Challenge**

The Image Classification Challenge:
1,000 object classes
1,431,167 images

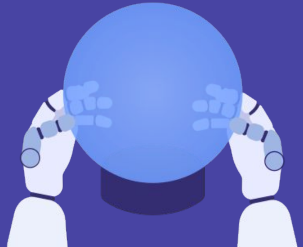
Output:
Scale
T-shirt
<u>Steel drum</u>
Drumstick
Mud turtle

✓

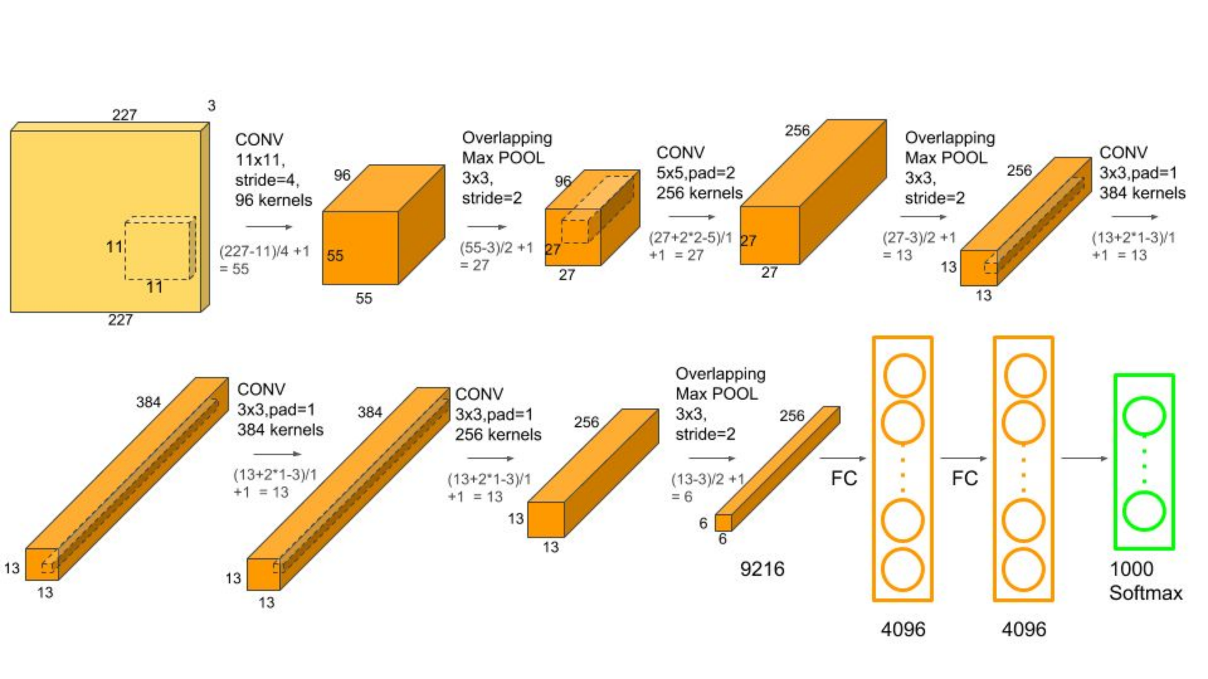
Output:
Scale
T-shirt
Giant panda
Drumstick
Mud turtle

✗

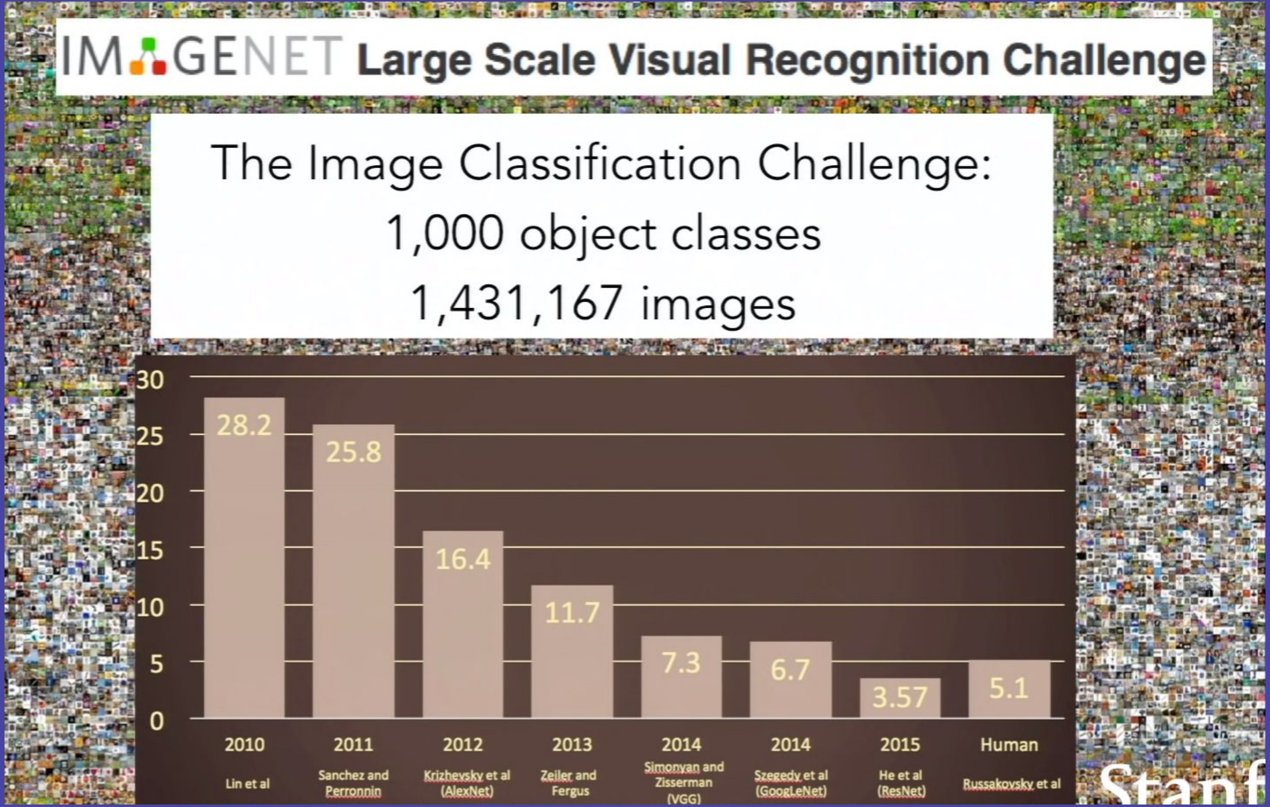
Stanf



AlexNet!


227
3
11
11
227
CONV
11x11,
stride=4,
96 kernels
(227-11)/4 +1
= 55
96
55
55
Overlapping
Max POOL
3x3,
stride=2
(55-3)/2 +1
= 27
96
27
27
CONV
5x5,pad=2
256 kernels
(27+2*2-5)/1
+1 = 27
256
27
27
Overlapping
Max POOL
3x3,
stride=2
(27-3)/2 +1
= 13
256
13
13
CONV
3x3,pad=1
384 kernels
(13+2*1-3)/1
+1 = 13
384
13
13
CONV
3x3,pad=1
384 kernels
(13+2*1-3)/1
+1 = 13
384
13
13
CONV
3x3,pad=1
256 kernels
(13+2*1-3)/1
+1 = 13
256
13
13
Overlapping
Max POOL
3x3,
stride=2
(13-3)/2 +1
= 6
256
6
6
9216
FC
4096
FC
4096
1000
Softmax

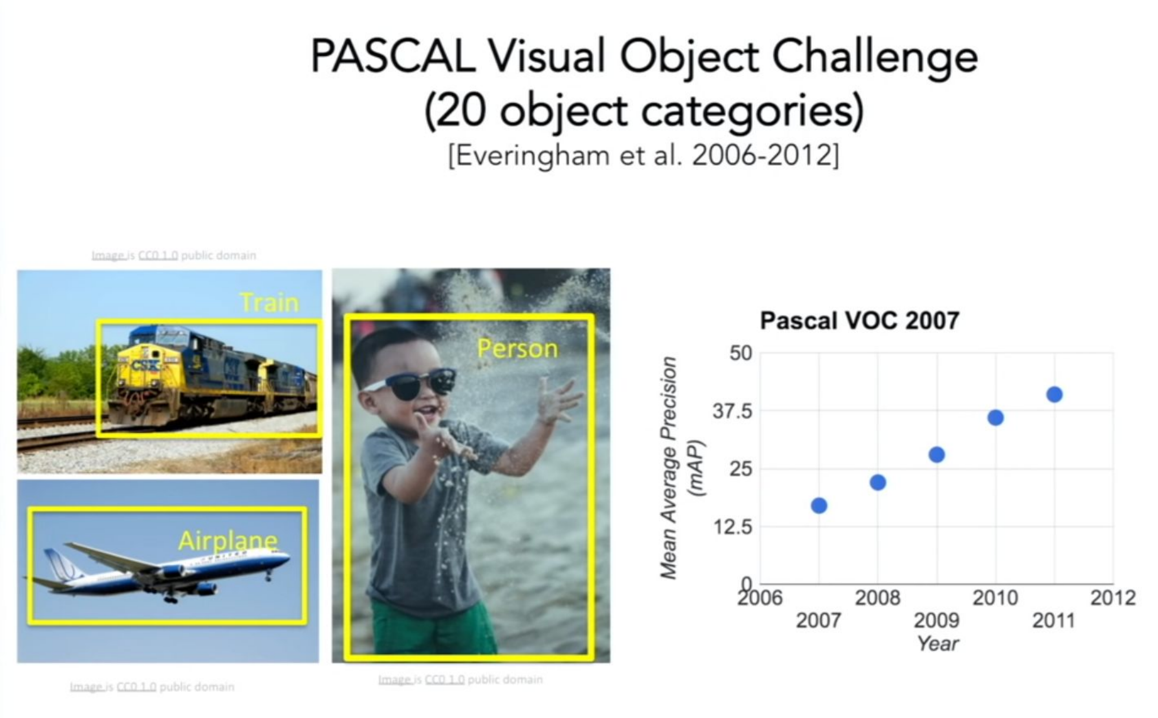
ImageNet progress



IMAGENET Large Scale Visual Recognition Challenge

The Image Classification Challenge:
1,000 object classes
1,431,167 images

Image Classification



PASCAL Visual Object Challenge
(20 object categories)
[Everingham et al. 2006-2012]

Image is CC0 1.0 public domain

Image is CC0 1.0 public domain

Image is CC0 1.0 public domain

Why did we only start using Neural Networks in 2012?

Cat vs dog in CNNs

Understanding the task

37 classes of dog and cats breeds

What accuracy can you get?

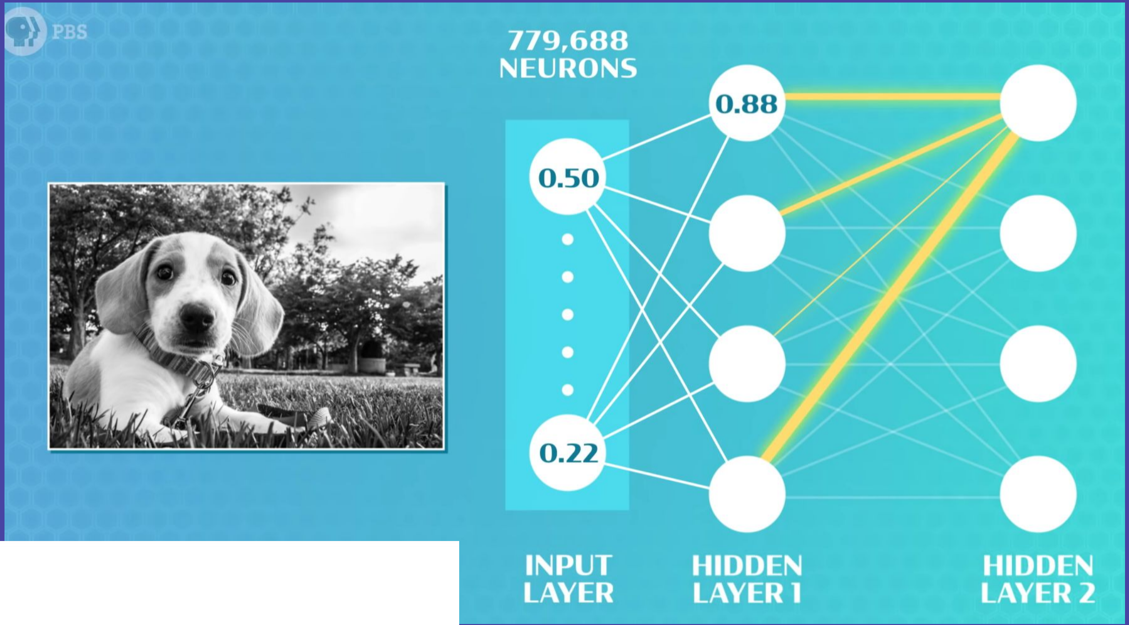
People with no idea about AI, telling me my AI will destroy the world
Me wondering why my neural network is classifying a cat as a dog..
Dog

Setting up the lab

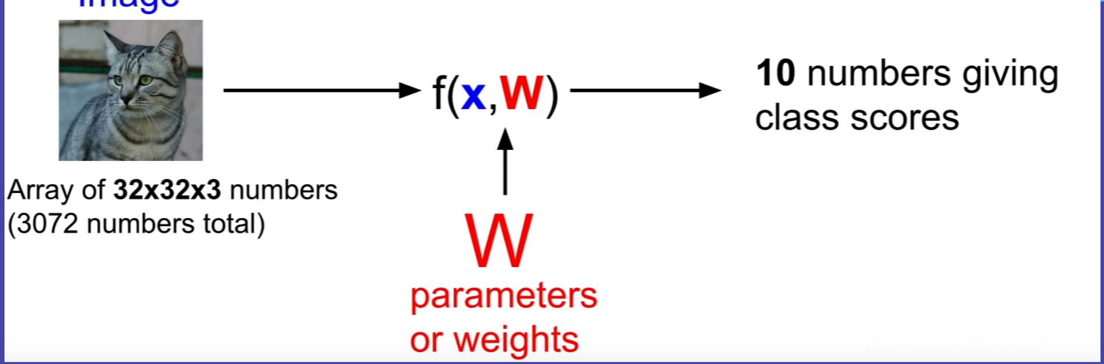
http://tiny.cc/des2020nnlab

python™ jupyter colab

Idea



Image

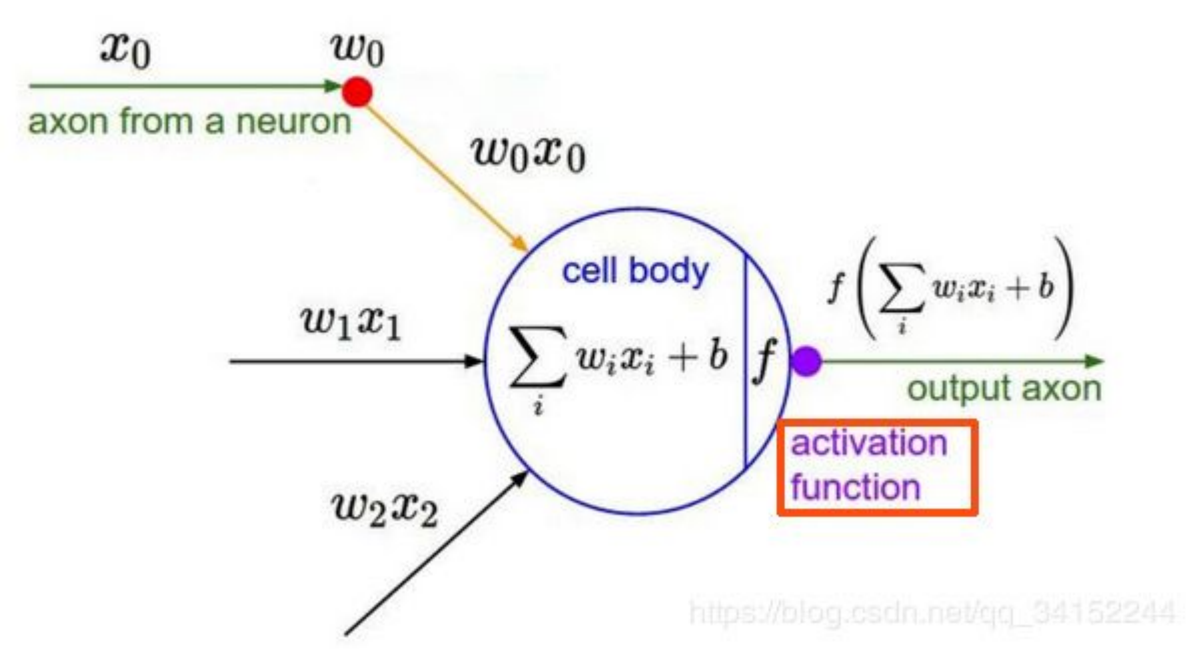


Array of **32x32x3** numbers
(3072 numbers total)

10 numbers giving
class scores

parameters
or weights

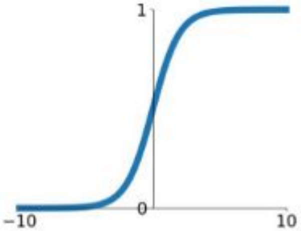
Neuron


x_0
w_0
axon from a neuron
w_0x_0
cell body
w_1x_1
$\sum_i w_ix_i + b$
f
$f\left(\sum_i w_ix_i + b\right)$
output axon
activation function
w_2x_2
https://blog.csdn.net/qq_34152244

Activation function

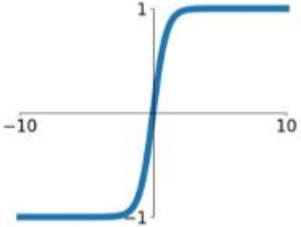
Sigmoid

$\sigma(x) = \frac{1}{1+e^{-x}}$



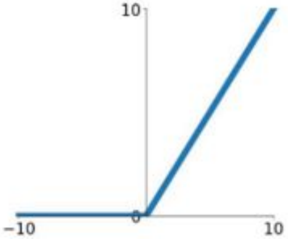
tanh

$\tanh(x)$



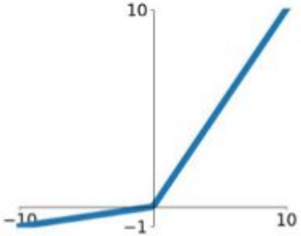
ReLU

$\max(0, x)$



Leaky ReLU

$\max(0.1x, x)$

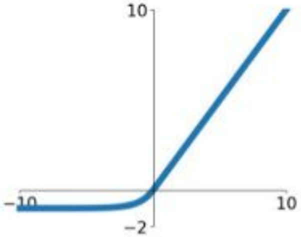


Maxout

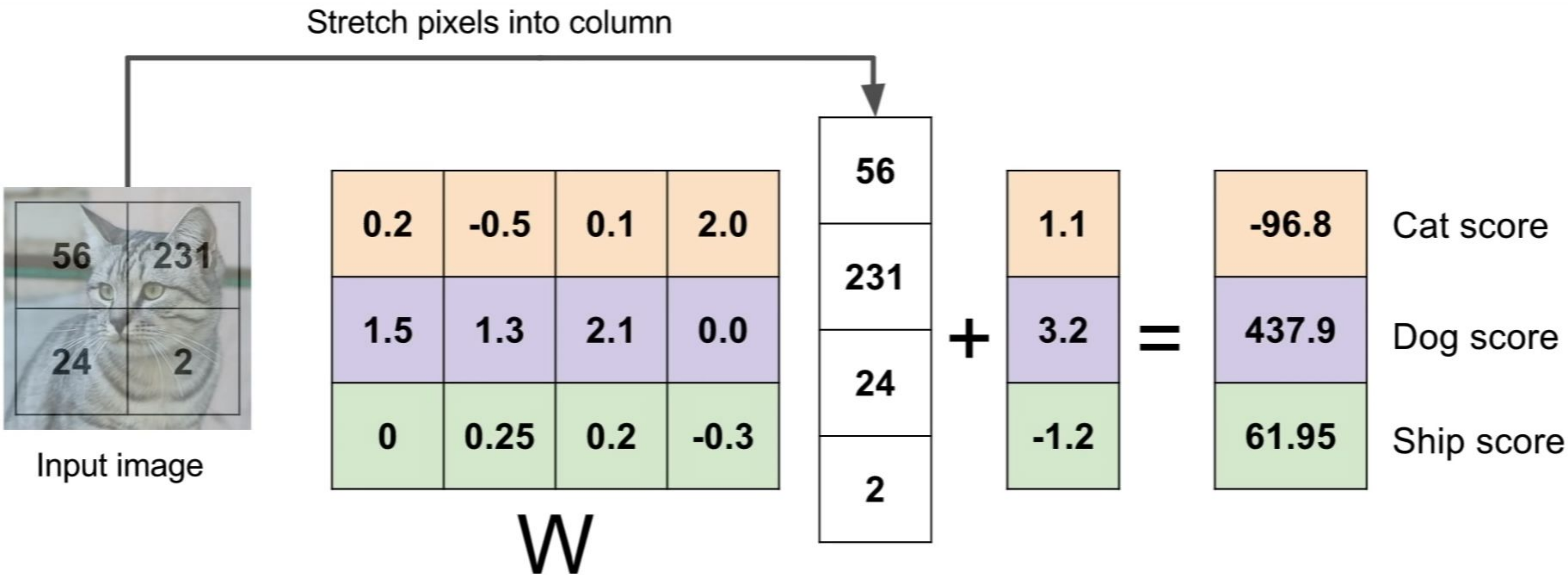
$\max(w_1^T x + b_1, w_2^T x + b_2)$

ELU

$$\begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$$



Fully connected layer

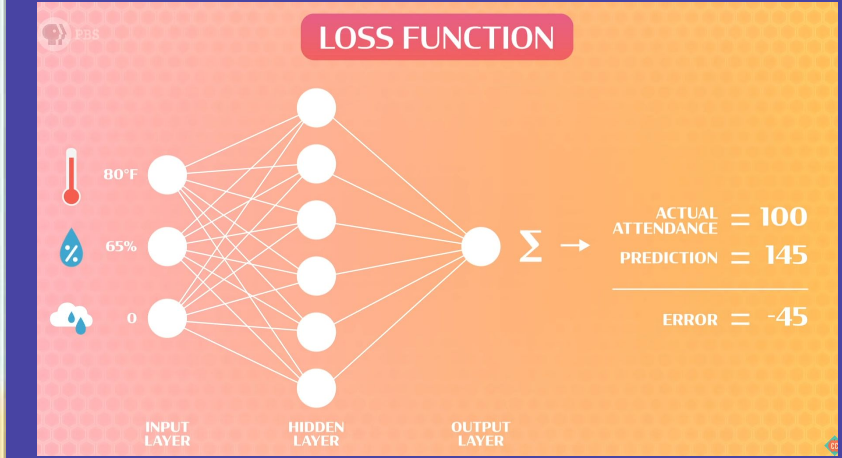


$$\underbrace{\begin{bmatrix} 0.2 & -0.5 & 0.1 & 2.0 \\ 1.5 & 1.3 & 2.1 & 0.0 \\ 0 & 0.25 & 0.2 & -0.3 \end{bmatrix}}_{W} \begin{bmatrix} 56 \\ 231 \\ 24 \\ 2 \end{bmatrix} + \begin{bmatrix} 1.1 \\ 3.2 \\ -1.2 \end{bmatrix} = \begin{array}{ll} \boxed{-96.8} & \text{Cat score} \\ \boxed{437.9} & \text{Dog score} \\ \boxed{61.95} & \text{Ship score} \end{array}$$

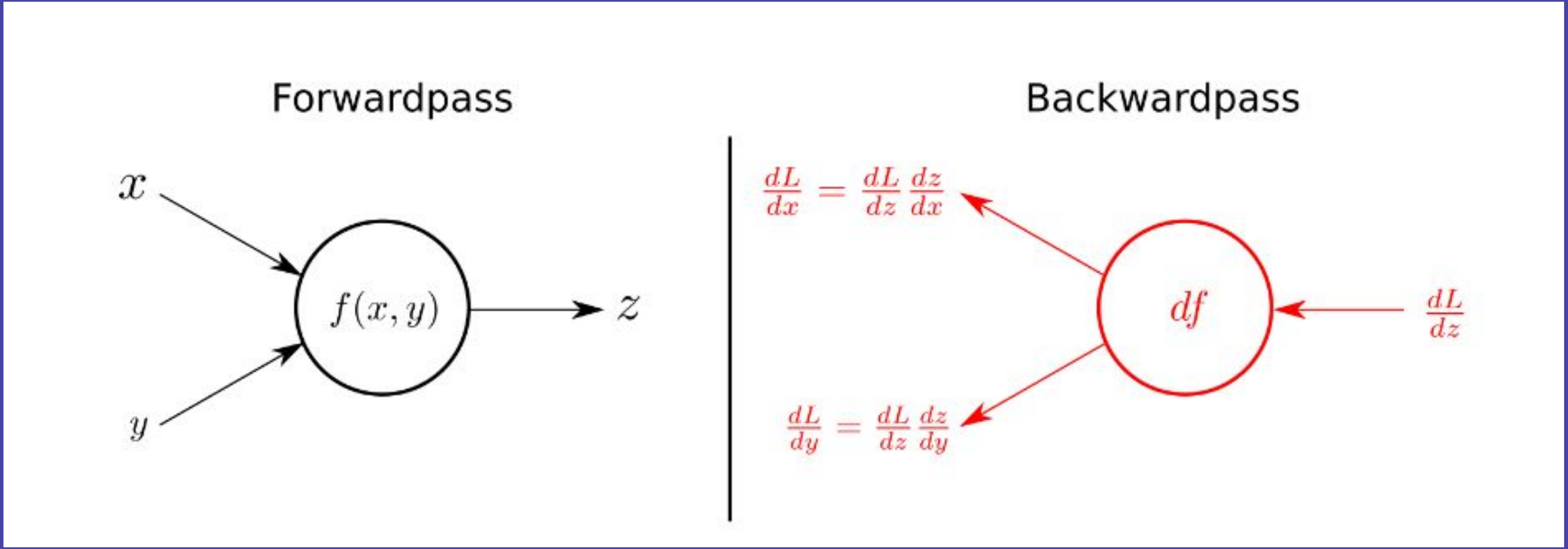
Training the network

Backpropagation

IT IS FRIDAY, ALL MY FRIENDS ARE AT PARTY
BUT, I'M HERE LEARNING BACK PROPAGATION
memegenerator.net



Backpropagation



Forwardpass
x
y
$f(x, y)$
z
Backwardpass
$\frac{dL}{dx} = \frac{dL}{dz}\frac{dz}{dx}$
$\frac{dL}{dy} = \frac{dL}{dz}\frac{dz}{dy}$
df
$\frac{dL}{dz}$

Backpropagation: a simple example

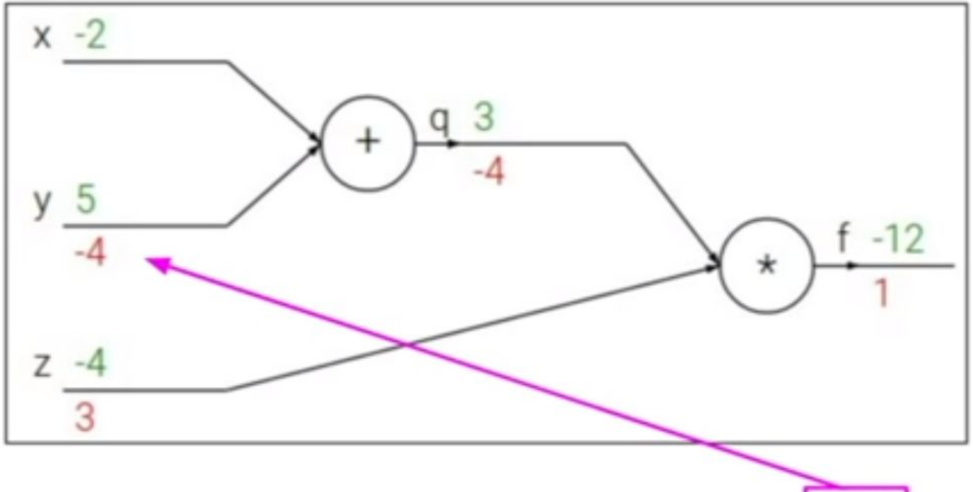
$$f(x, y, z) = (x + y)z$$

e.g. x = -2, y = 5, z = -4

$$q = x + y \qquad \frac{\partial q}{\partial x} = 1, \frac{\partial q}{\partial y} = 1$$

$$f = qz \qquad \frac{\partial f}{\partial q} = z, \frac{\partial f}{\partial z} = q$$

Want: $\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$

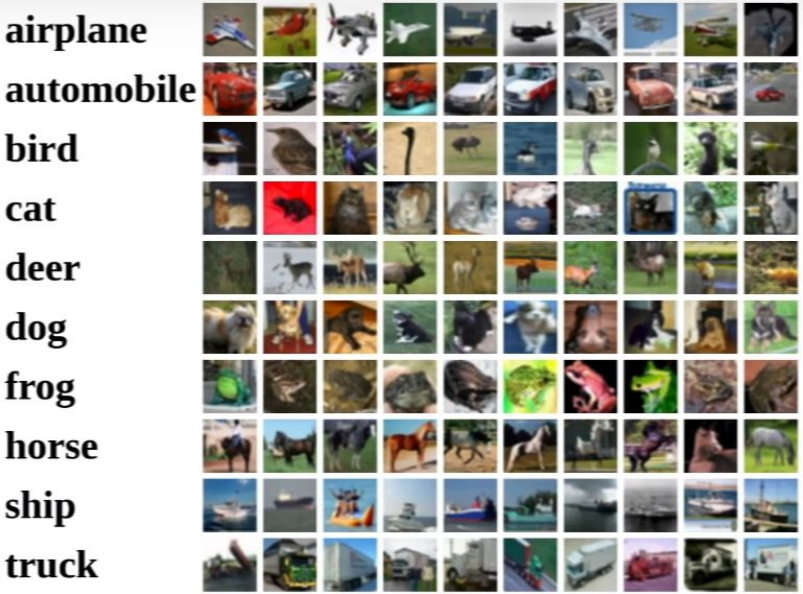


$$\frac{\partial f}{\partial y}$$

Chain rule:

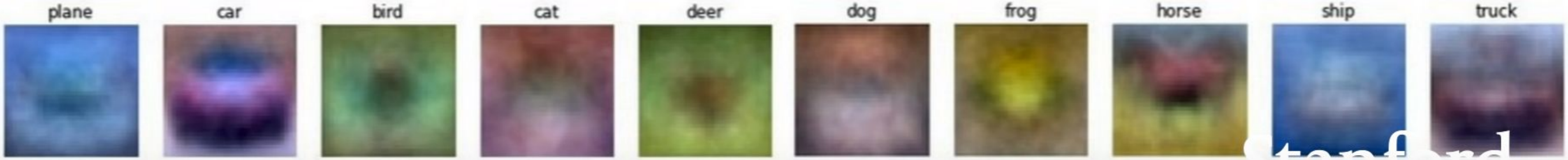
$$\boxed{\frac{\partial f}{\partial y} = \frac{\partial f}{\partial q} \frac{\partial q}{\partial y}}$$

NNs trained for CIFAR10 dataset

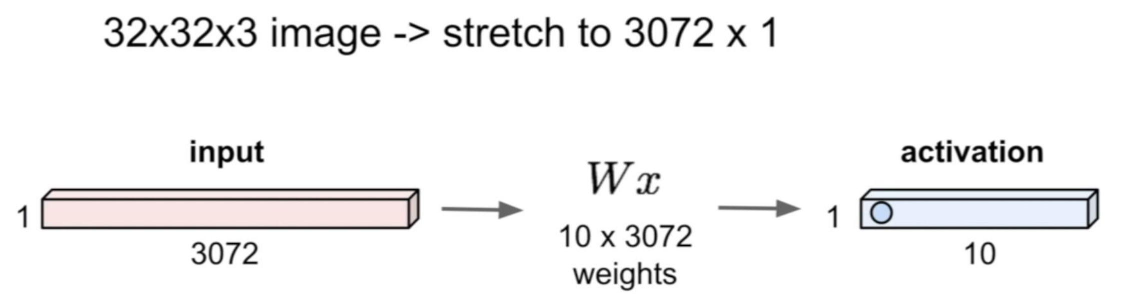


$$f(x,W) = Wx + b$$

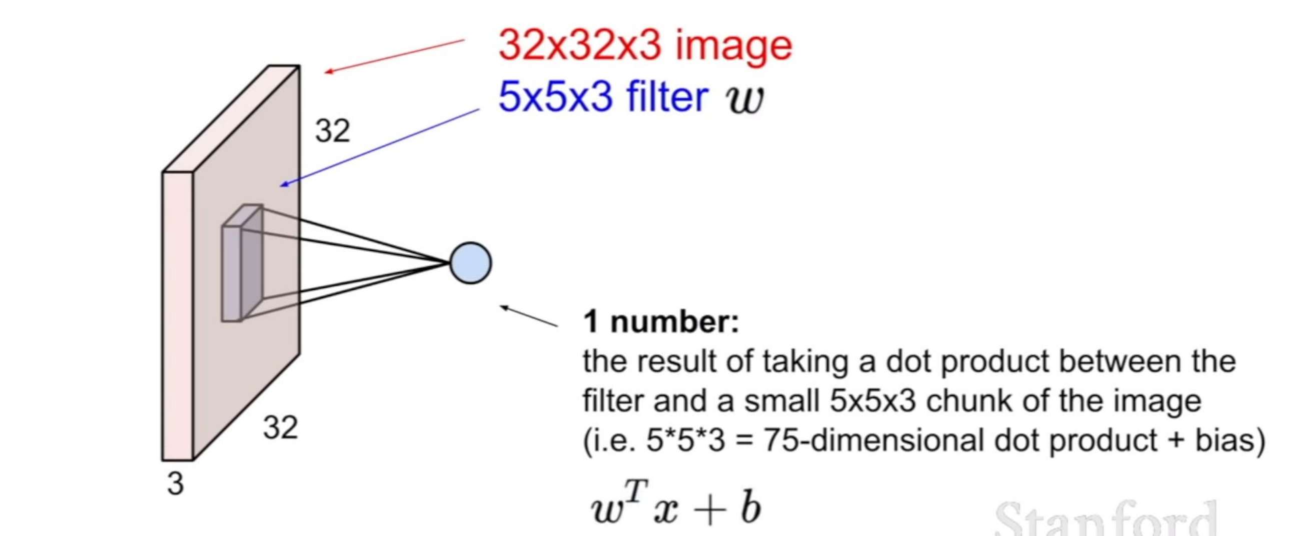
Example trained weights of a linear classifier trained on CIFAR-10:



Fully Connected vs Convolutional



32x32x3 image -> stretch to 3072 x 1

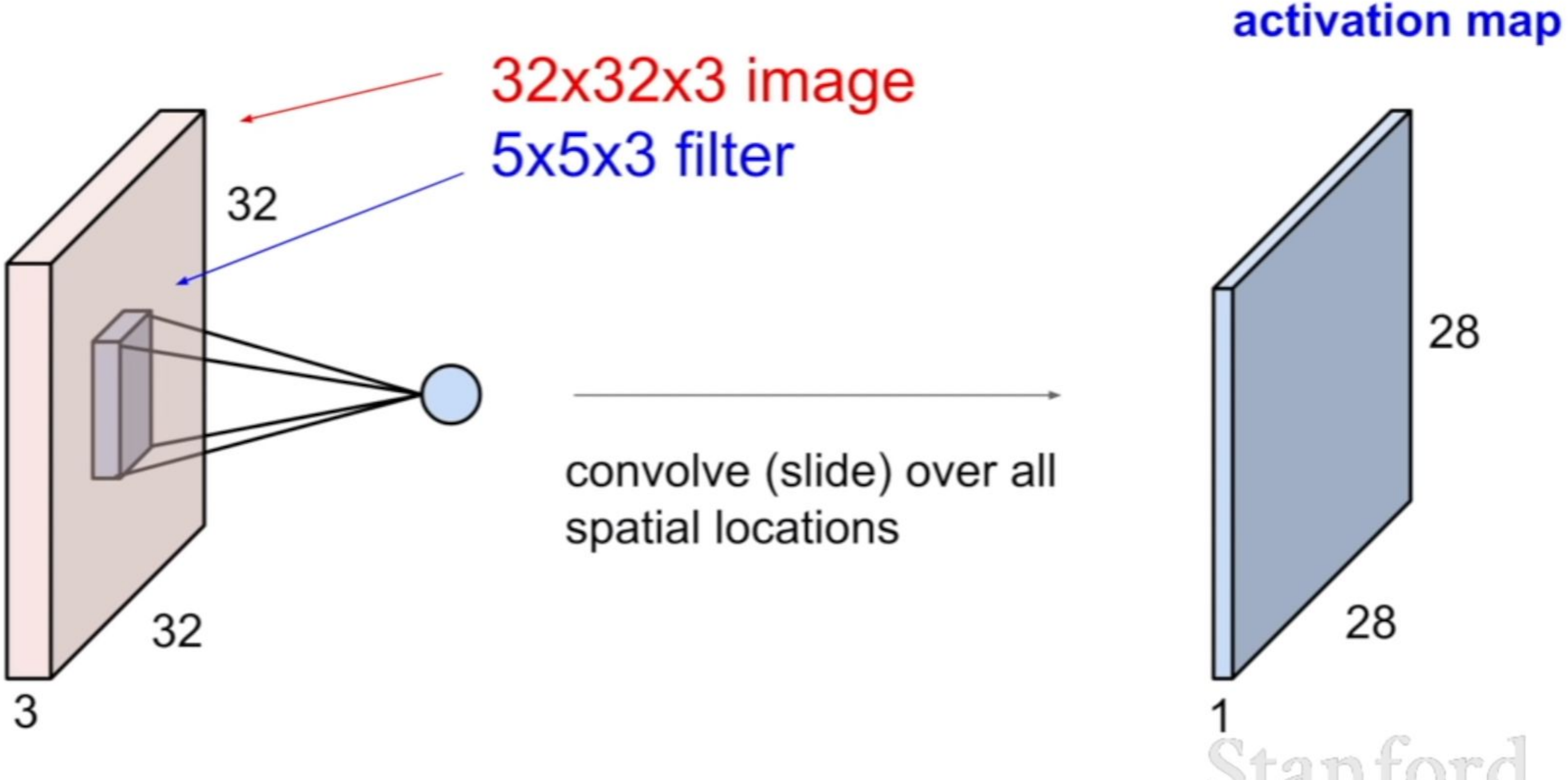


1 number:
the result of taking a dot product between the filter and a small 5x5x3 chunk of the image (i.e. 5*5*3 = 75-dimensional dot product + bias)

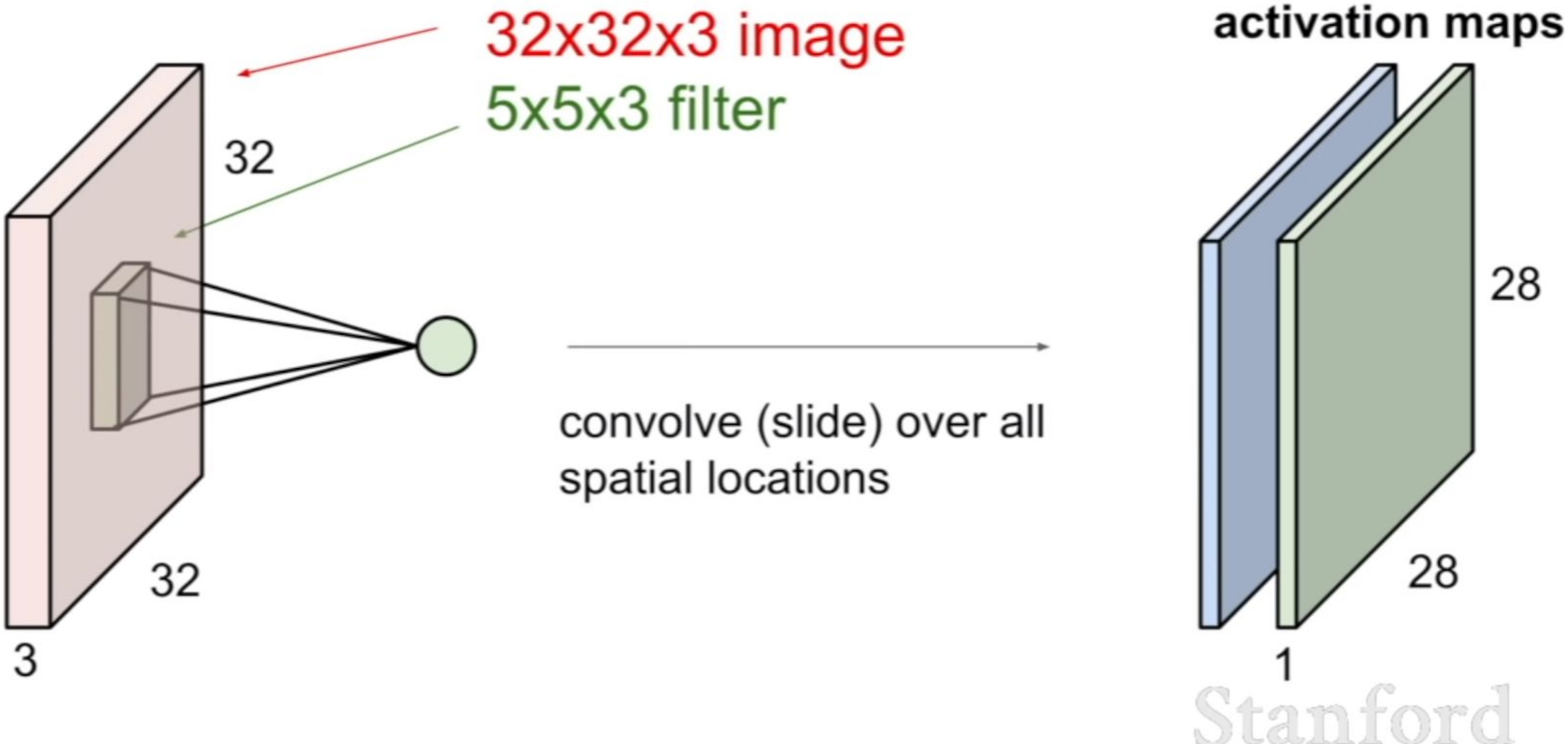
$$w^T x + b$$

Filters

Filters


activation map
32x32x3 image
5x5x3 filter
32
32
3
convolve (slide) over all spatial locations
28
28
1

Filters



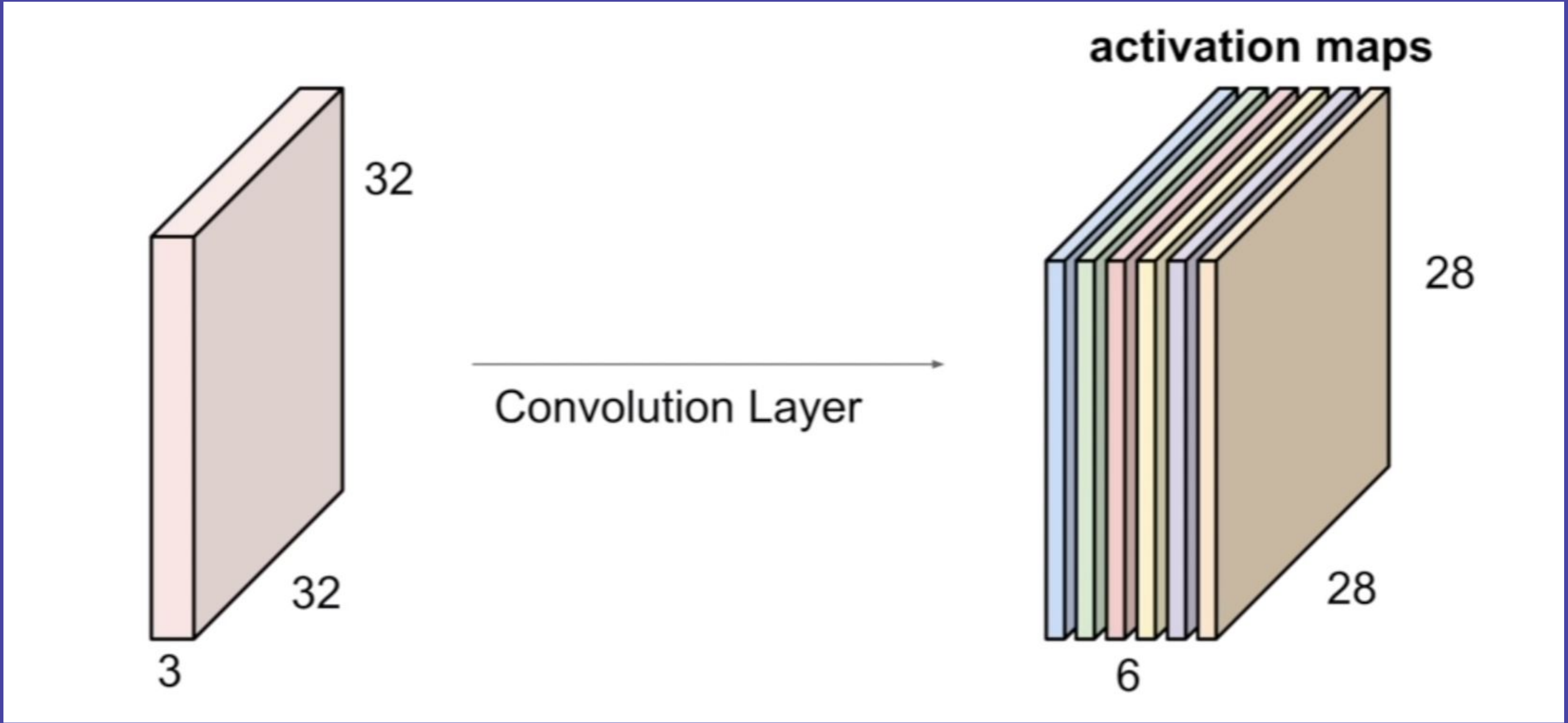
32x32x3 image

5x5x3 filter

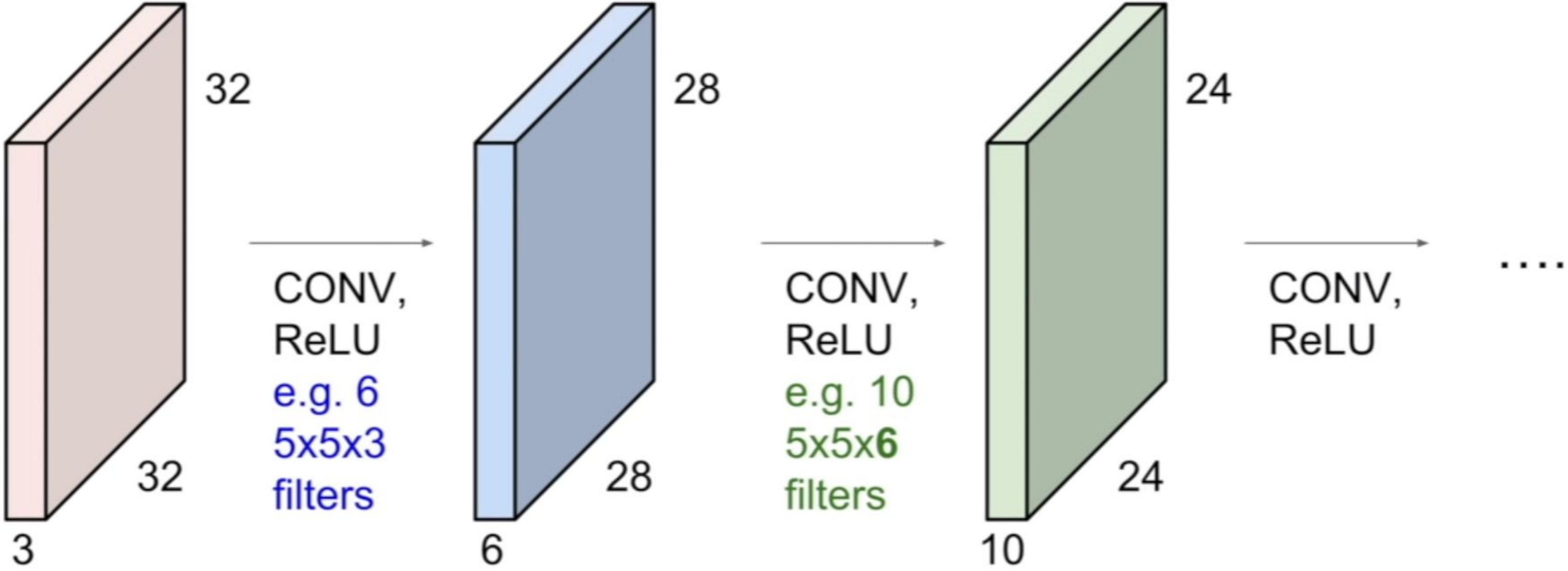
convolve (slide) over all spatial locations

activation maps

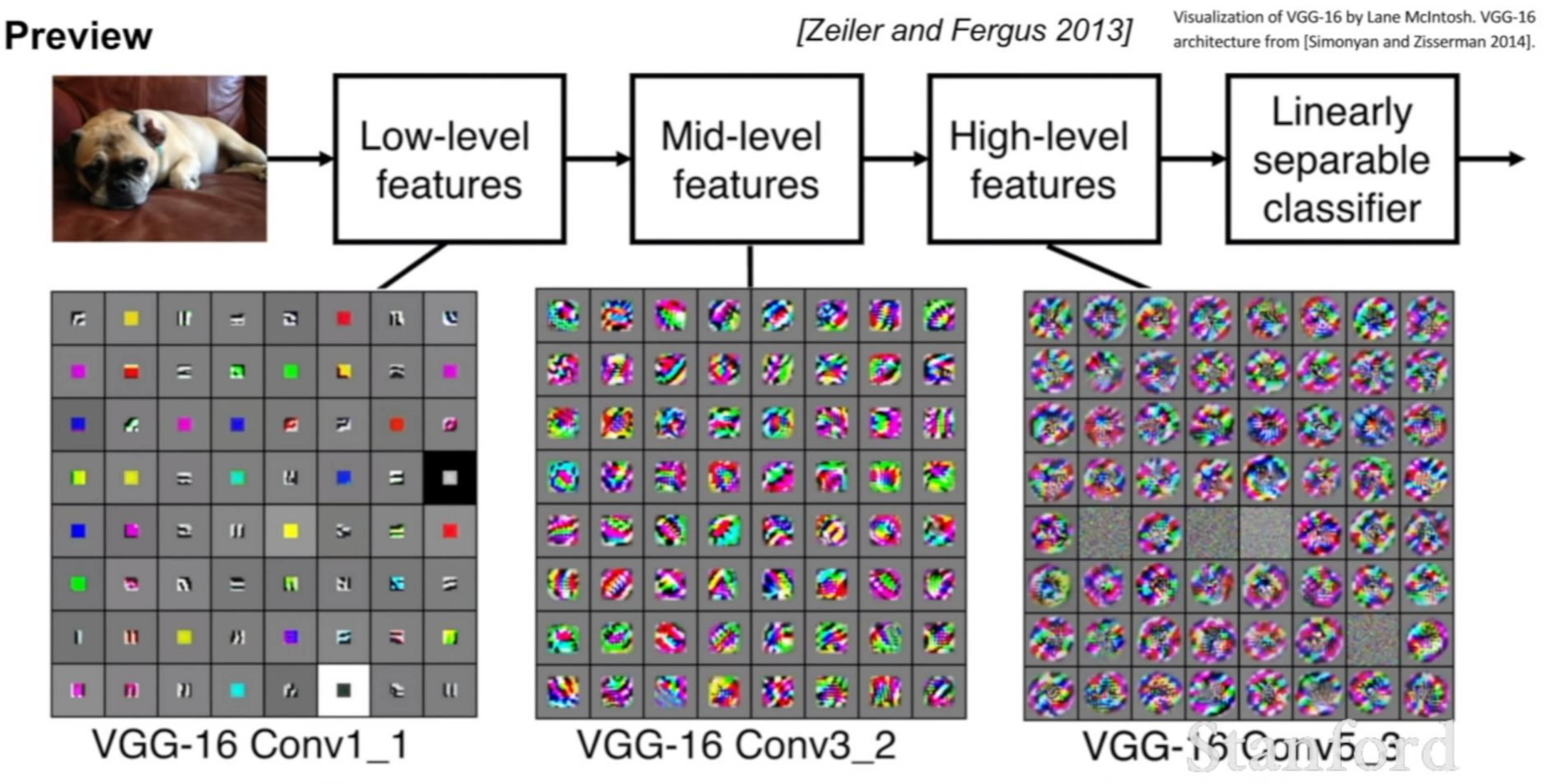
Filters



Network


32
32
3
CONV,
ReLU
e.g. 6
5x5x3
filters
28
28
6
CONV,
ReLU
e.g. 10
5x5x**6**
filters
24
24
10
CONV,
ReLU
....

Understanding the layers



Preview

[Zeiler and Fergus 2013]

Visualization of VGG-16 by Lane McIntosh. VGG-16 architecture from [Simonyan and Zisserman 2014].

Low-level features → Mid-level features → High-level features → Linearly separable classifier →

VGG-16 Conv1_1

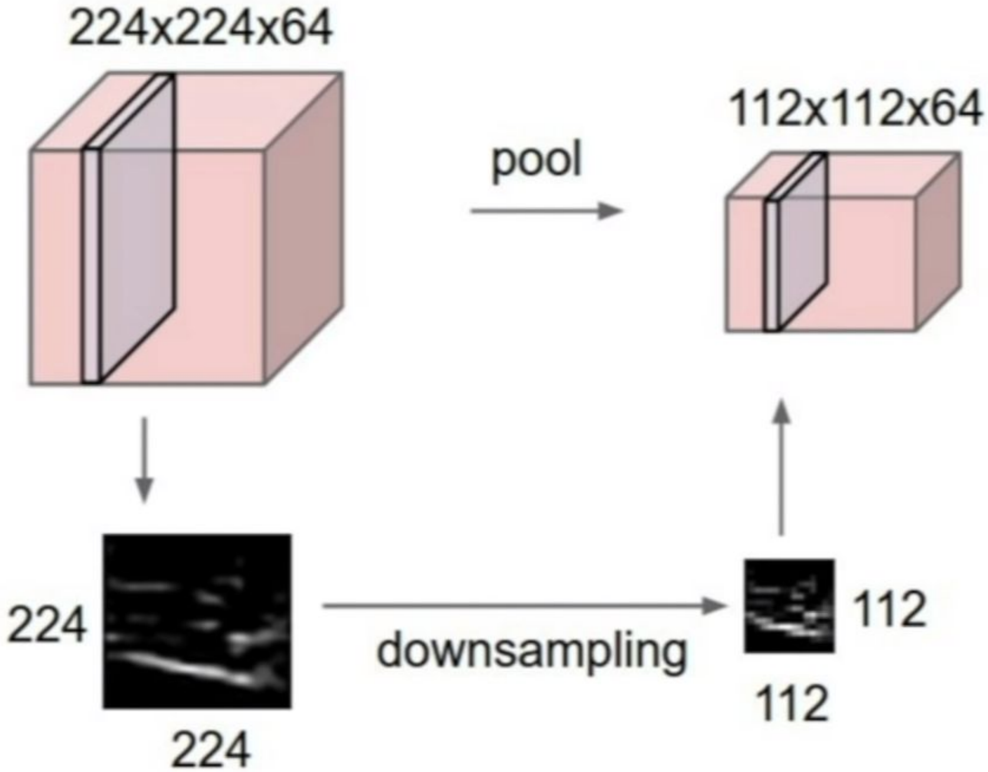
VGG-16 Conv3_2

VGG-16 Conv5_3

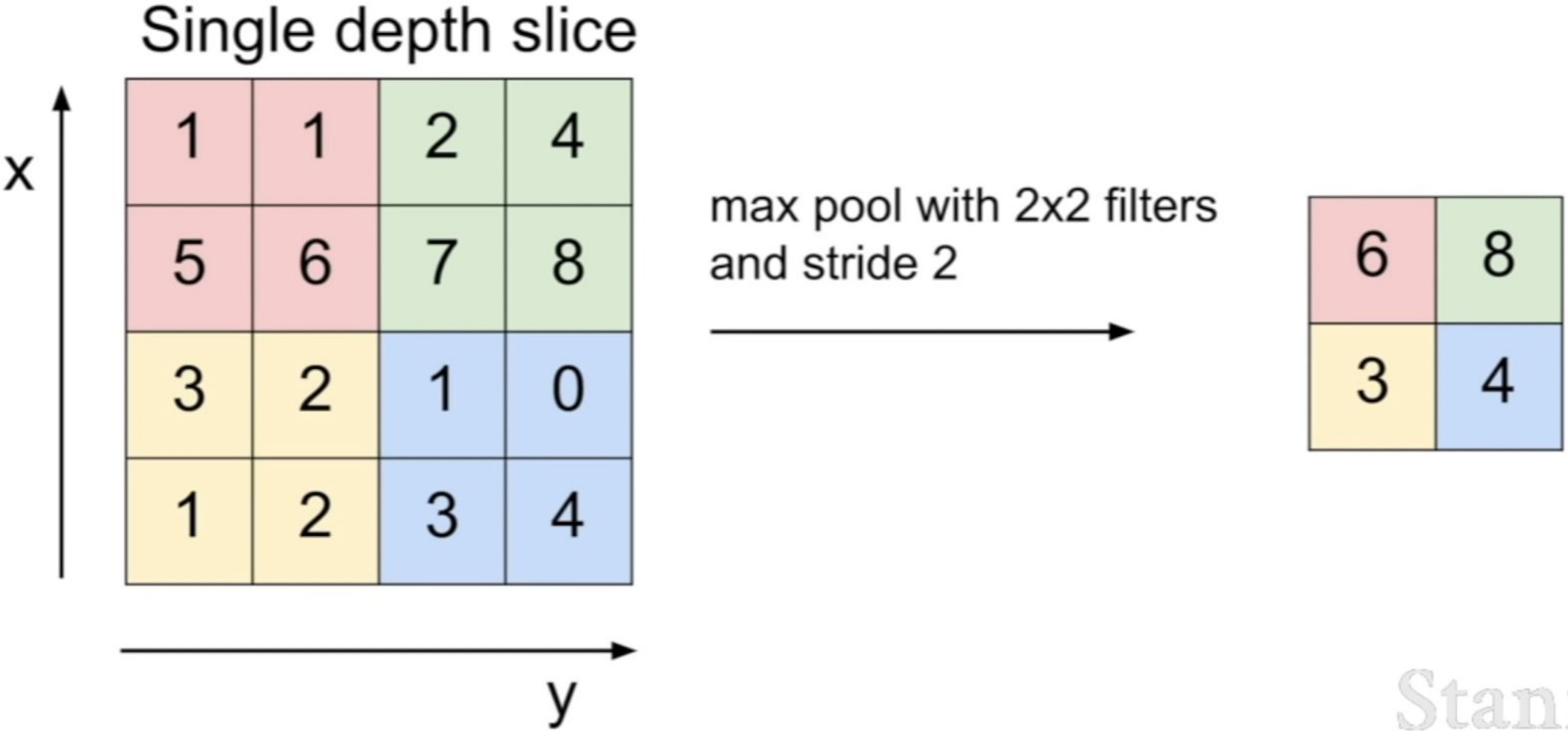
Pooling

Pooling layer

- makes the representations smaller and more manageable
- operates over each activation map independently:



MAX POOLING

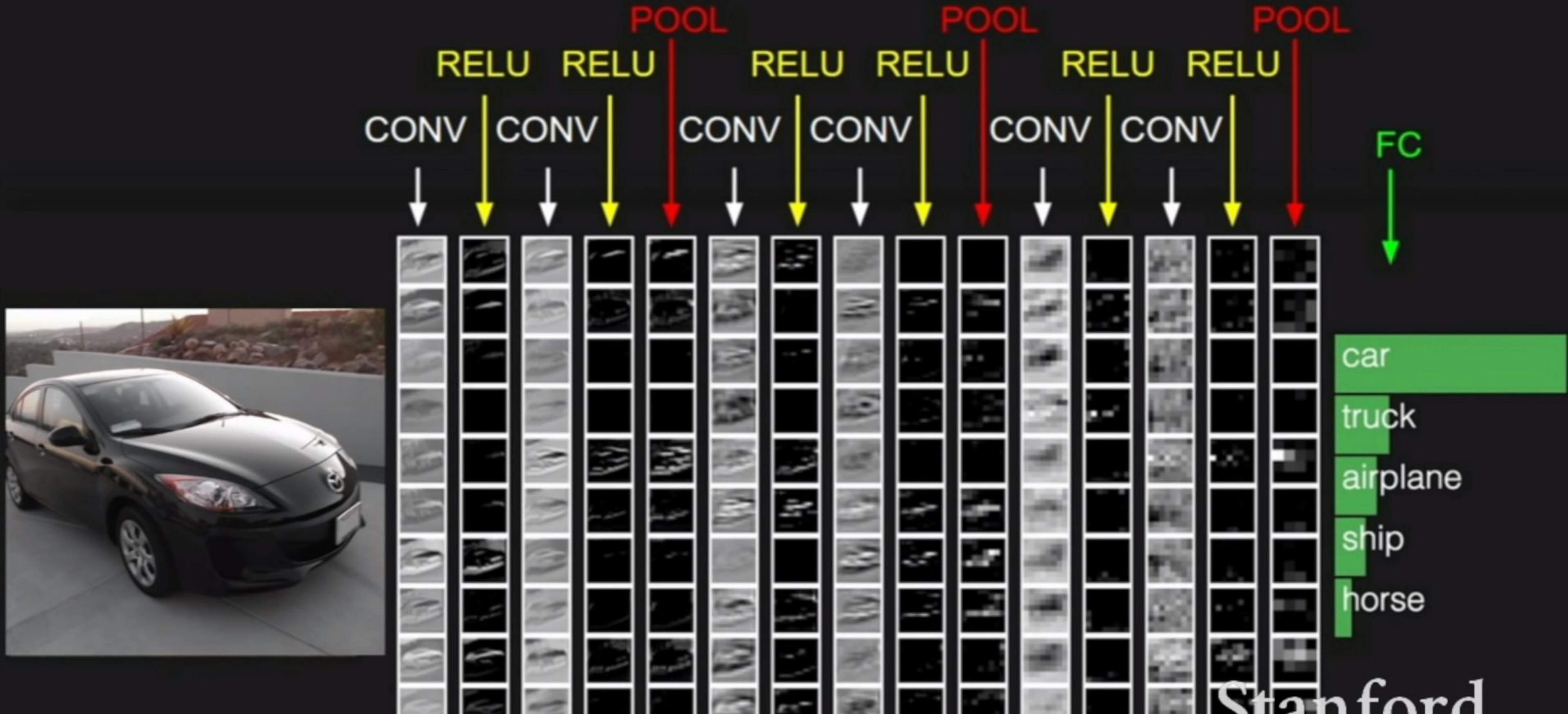


Single depth slice

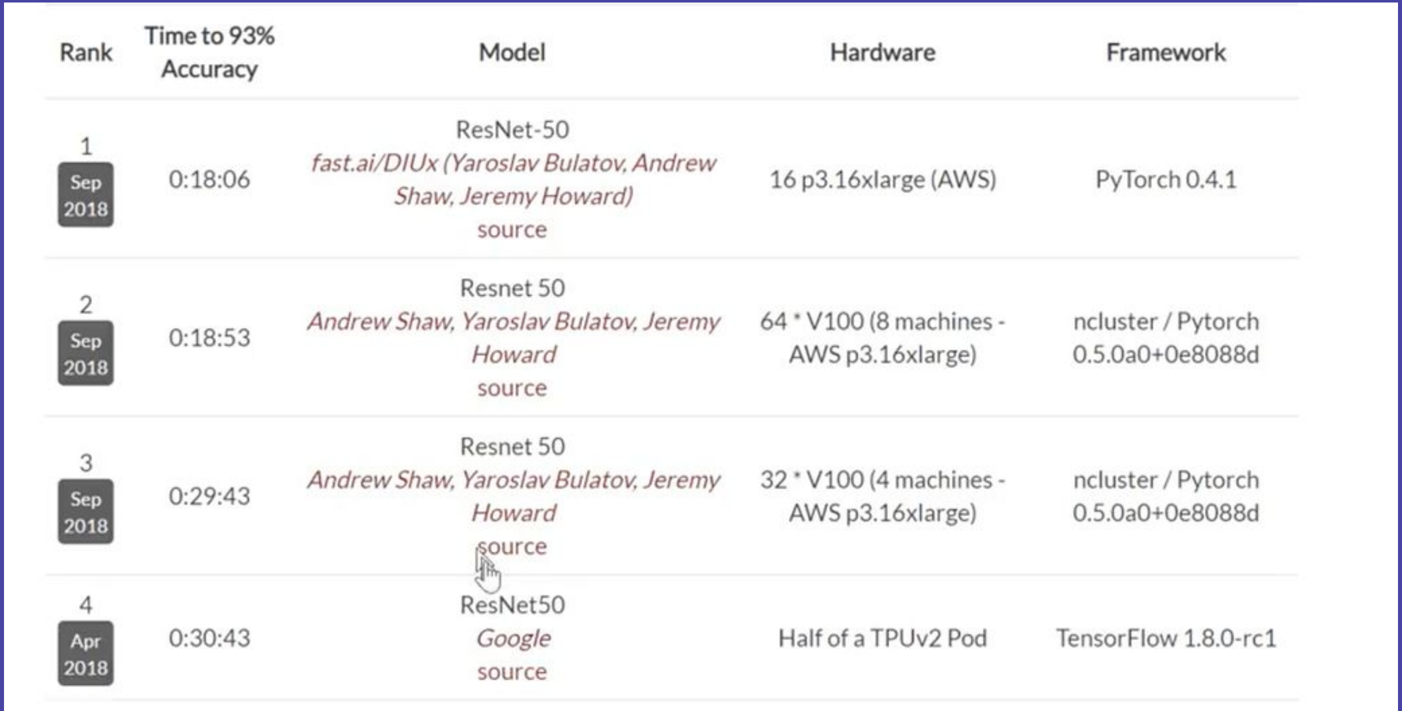
1	1	2	4
5	6	7	8
3	2	1	0
1	2	3	4

max pool with 2x2 filters and stride 2

6	8
3	4

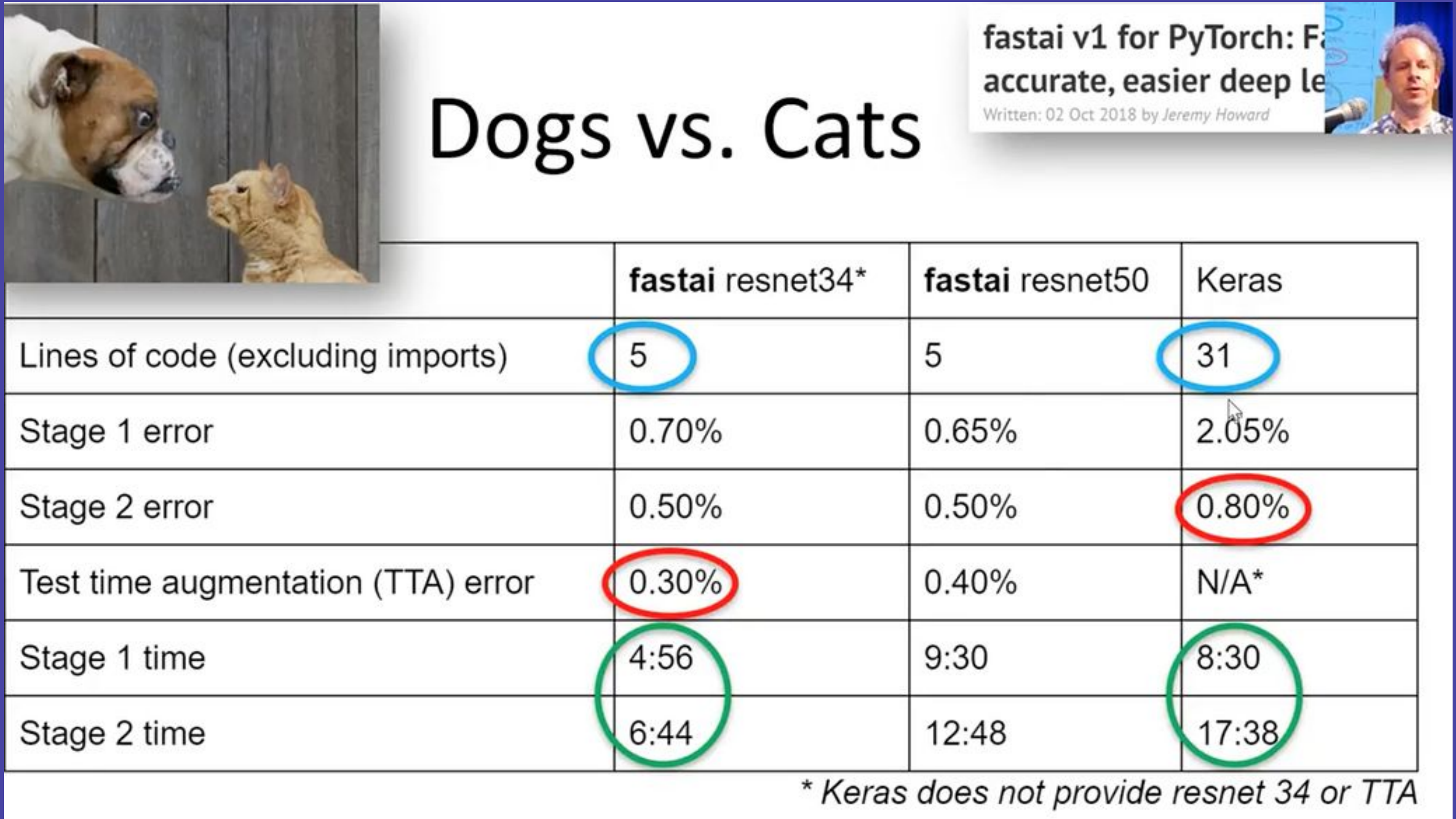

POOL
POOL
POOL
RELU
RELU
RELU
RELU
RELU
RELU
CONV
CONV
CONV
CONV
CONV
CONV
FC
car
truck
airplane
ship
horse

Why ResNet?



Rank	Time to 93% Accuracy	Model	Hardware	Framework
1 Sep 2018	0:18:06	ResNet-50 *fast.ai/DIUx (Yaroslav Bulatov, Andrew Shaw, Jeremy Howard)* source	16 p3.16xlarge (AWS)	PyTorch 0.4.1
2 Sep 2018	0:18:53	Resnet 50 *Andrew Shaw, Yaroslav Bulatov, Jeremy Howard* source	64 * V100 (8 machines - AWS p3.16xlarge)	ncluster / Pytorch 0.5.0a0+0e8088d
3 Sep 2018	0:29:43	Resnet 50 *Andrew Shaw, Yaroslav Bulatov, Jeremy Howard* source	32 * V100 (4 machines - AWS p3.16xlarge)	ncluster / Pytorch 0.5.0a0+0e8088d
4 Apr 2018	0:30:43	ResNet50 *Google* source	Half of a TPUv2 Pod	TensorFlow 1.8.0-rc1

Other frameworks?



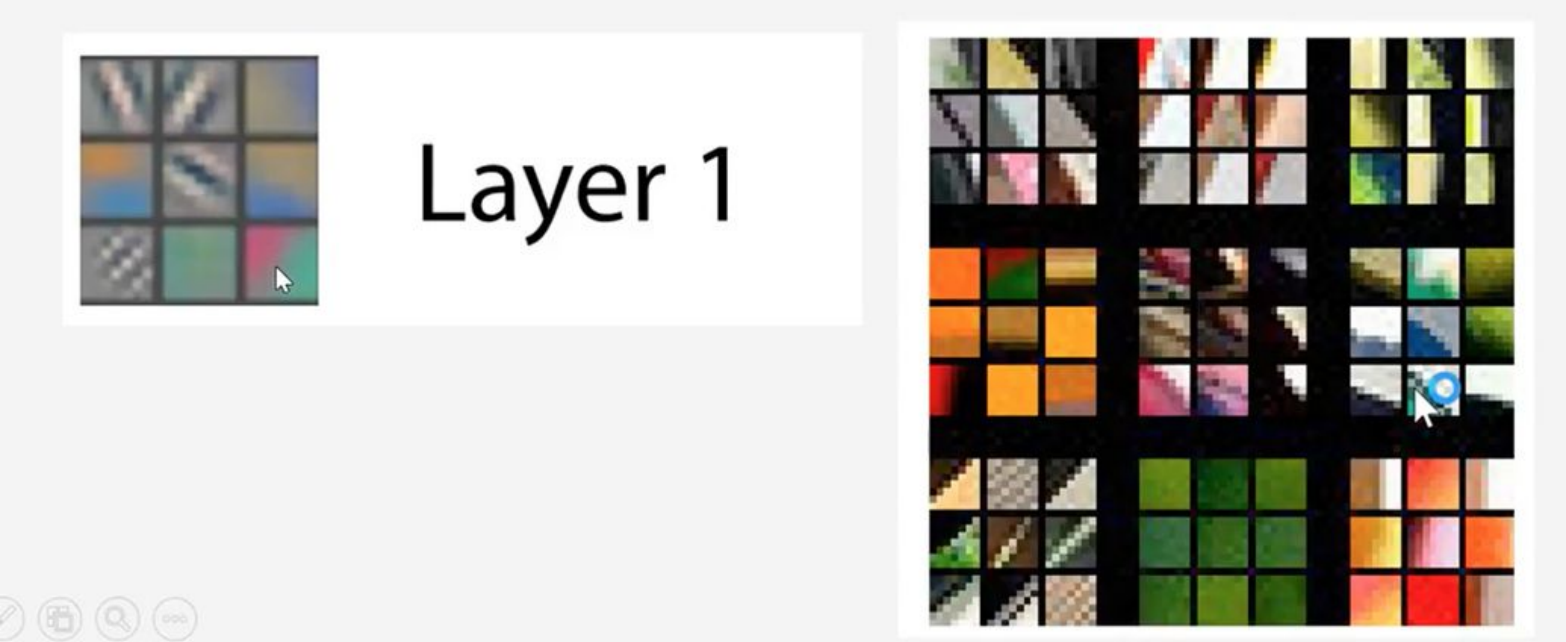
Dogs vs. Cats

fastai v1 for PyTorch: F
accurate, easier deep le
Written: 02 Oct 2018 by *Jeremy Howard*

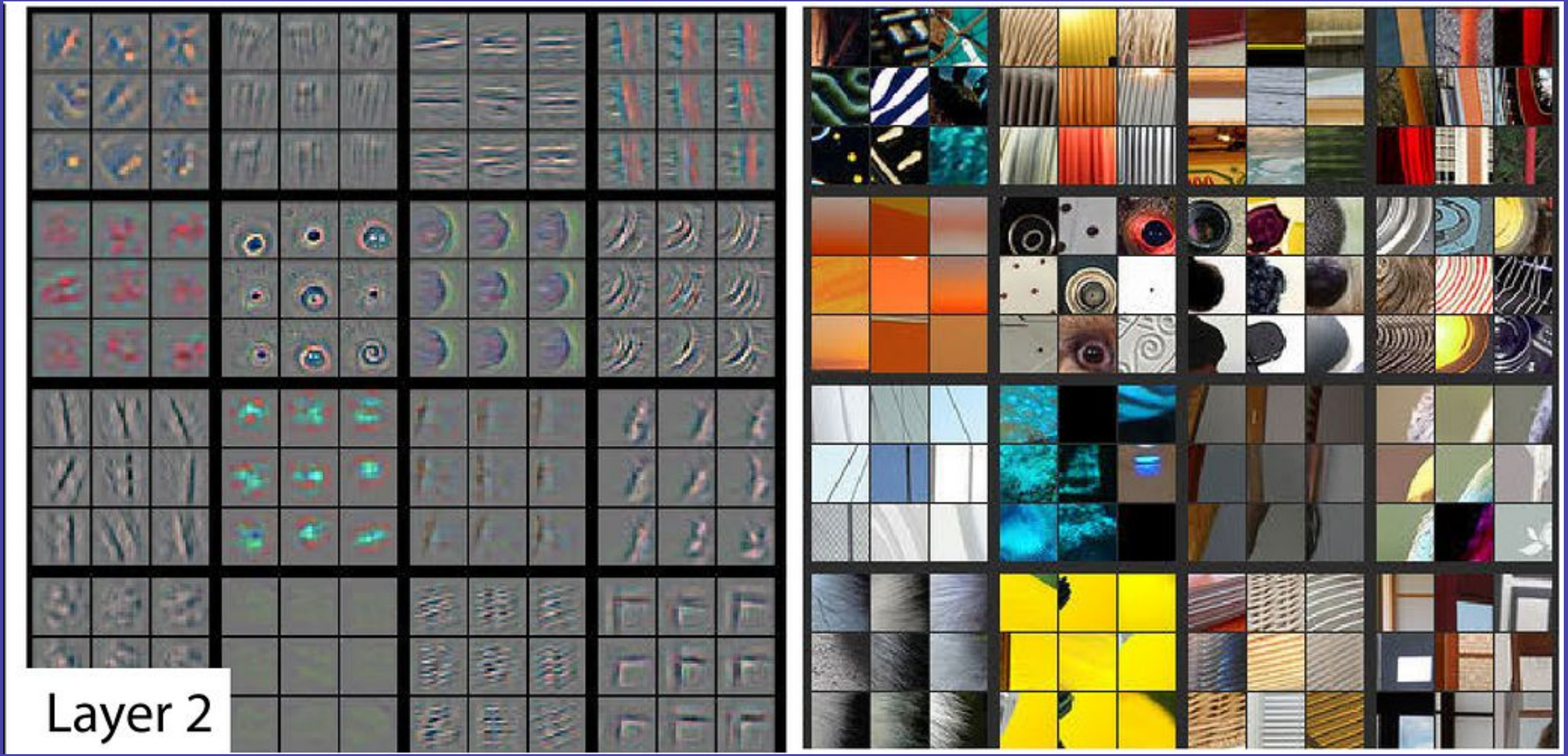
	fastai resnet34*	**fastai** resnet50	Keras
Lines of code (excluding imports)	5	5	31
Stage 1 error	0.70%	0.65%	2.05%
Stage 2 error	0.50%	0.50%	0.80%
Test time augmentation (TTA) error	0.30%	0.40%	N/A*
Stage 1 time	4:56	9:30	8:30
Stage 2 time	6:44	12:48	17:38

** Keras does not provide resnet 34 or TTA*

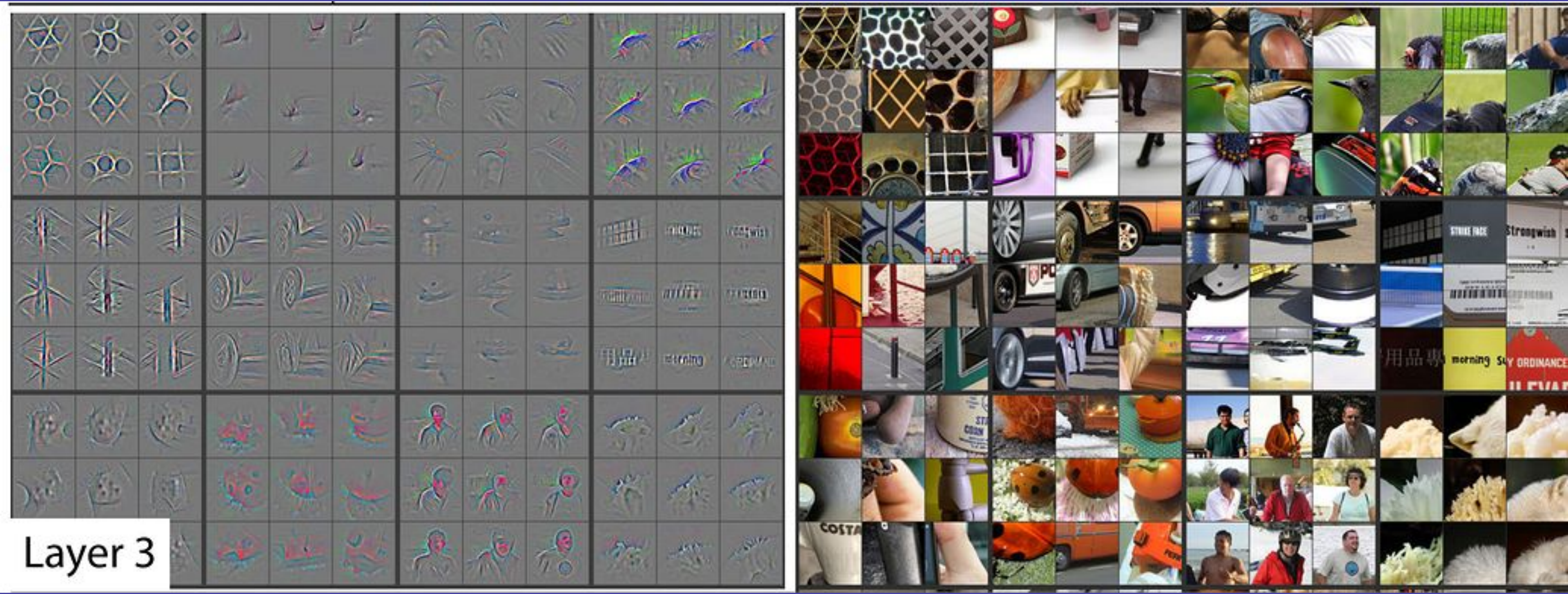
Understanding Conv Layers



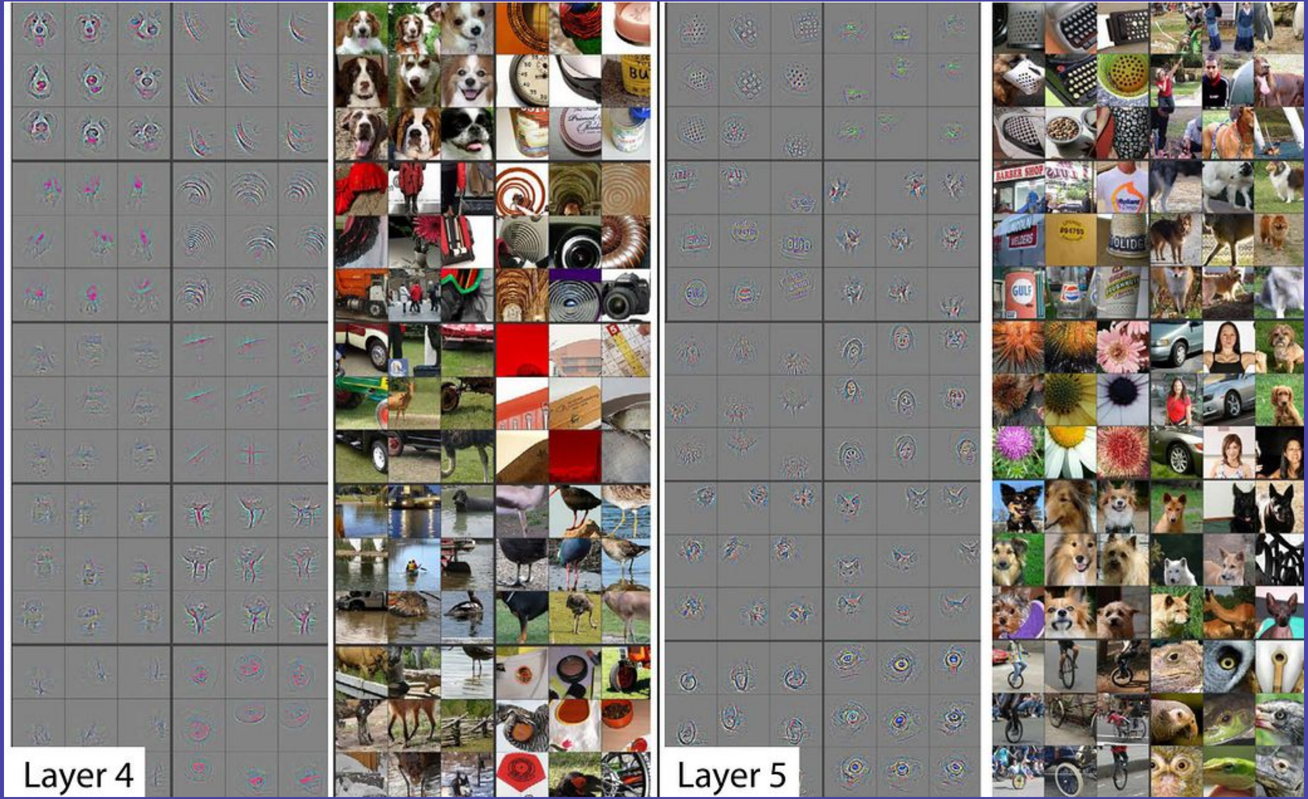
Layer 2



Layer 3

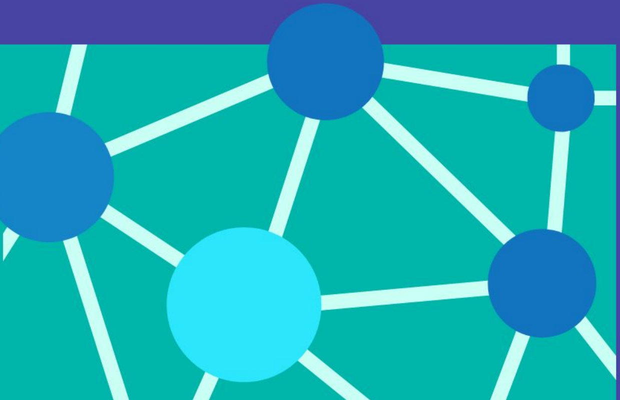


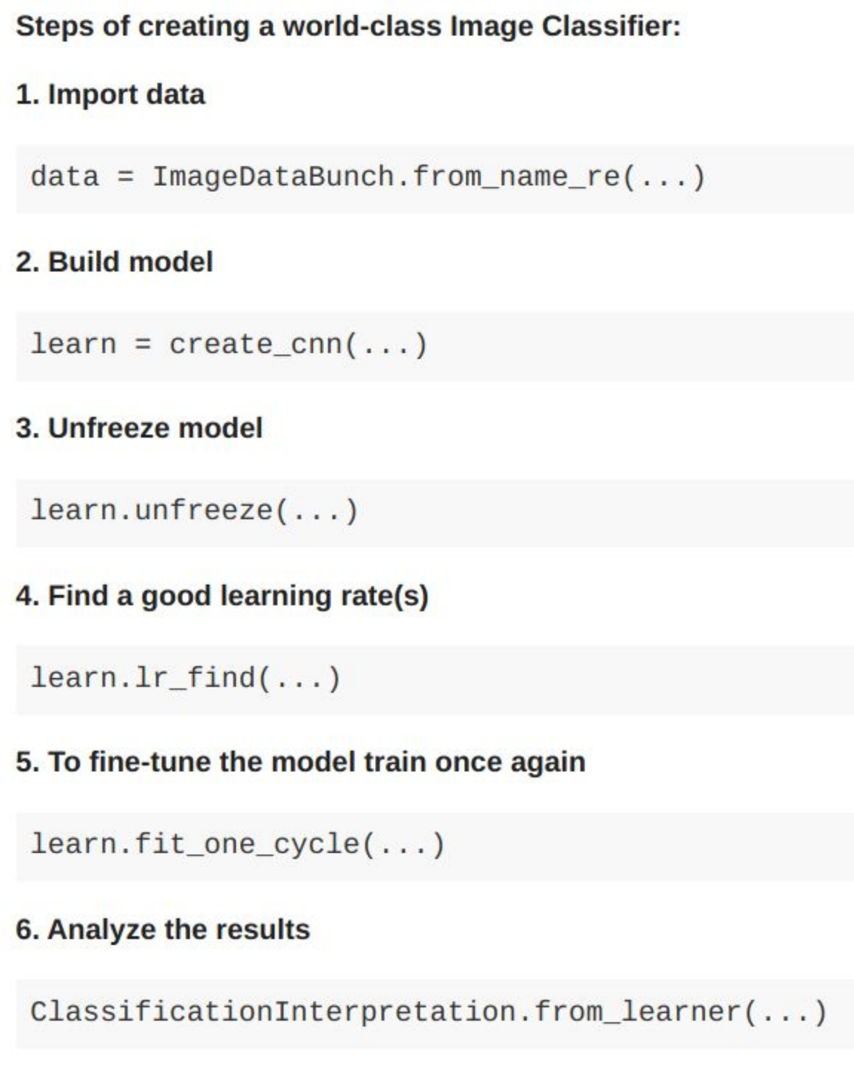
Layer 4 & 5



What’s next

Try to classify MNIST dataset ;)





Steps of creating a world-class Image Classifier:

1. Import data

```
data = ImageDataBunch.from_name_re(...)
```

2. Build model

```
learn = create_cnn(...)
```

3. Unfreeze model

```
learn.unfreeze(...)
```

4. Find a good learning rate(s)

```
learn.lr_find(...)
```

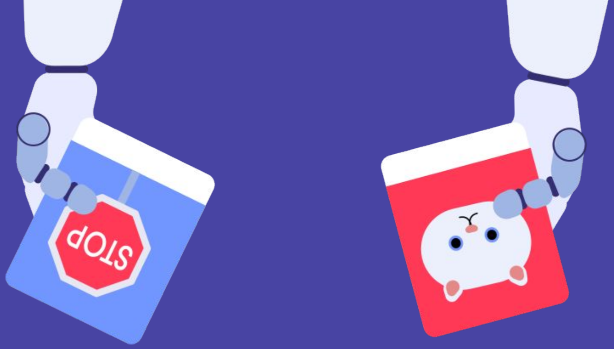
5. To fine-tune the model train once again

```
learn.fit_one_cycle(...)
```

6. Analyze the results

```
ClassificationInterpretation.from_learner(...)
```

Resources



Fast.ai course: https://course.fast.ai/

Andrew Ng courses: https://www.coursera.org/learn/machine-learning
https://www.youtube.com/watch?v=PySo_6S4ZAg&list=PLoROMvodv4rOABXSygHTsbvUz4G_YQhOb

Stanford CNN course:
https://www.youtube.com/watch?v=vT1JzLTH4G4&list=PL3FW7Lu3i5JvHM8ljYj-zLfQRF3EO8sYv

PyTorch course on Udacity:
https://www.udacity.com/course/deep-learning-pytorch--ud188