Deep learning

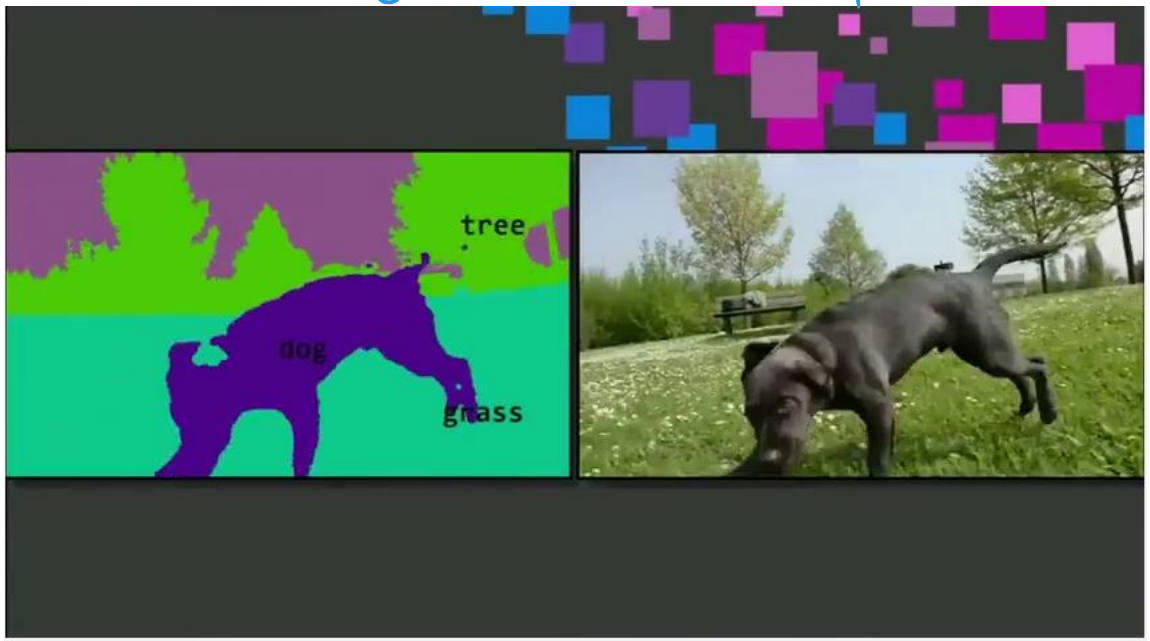
Object and activity recognition

Click to go to the video in Youtube

Large-scale Video Classification with Convolutional Neural Networks, CVPR 2014

Object detection and segmentation

Click to go to the video in Youtube



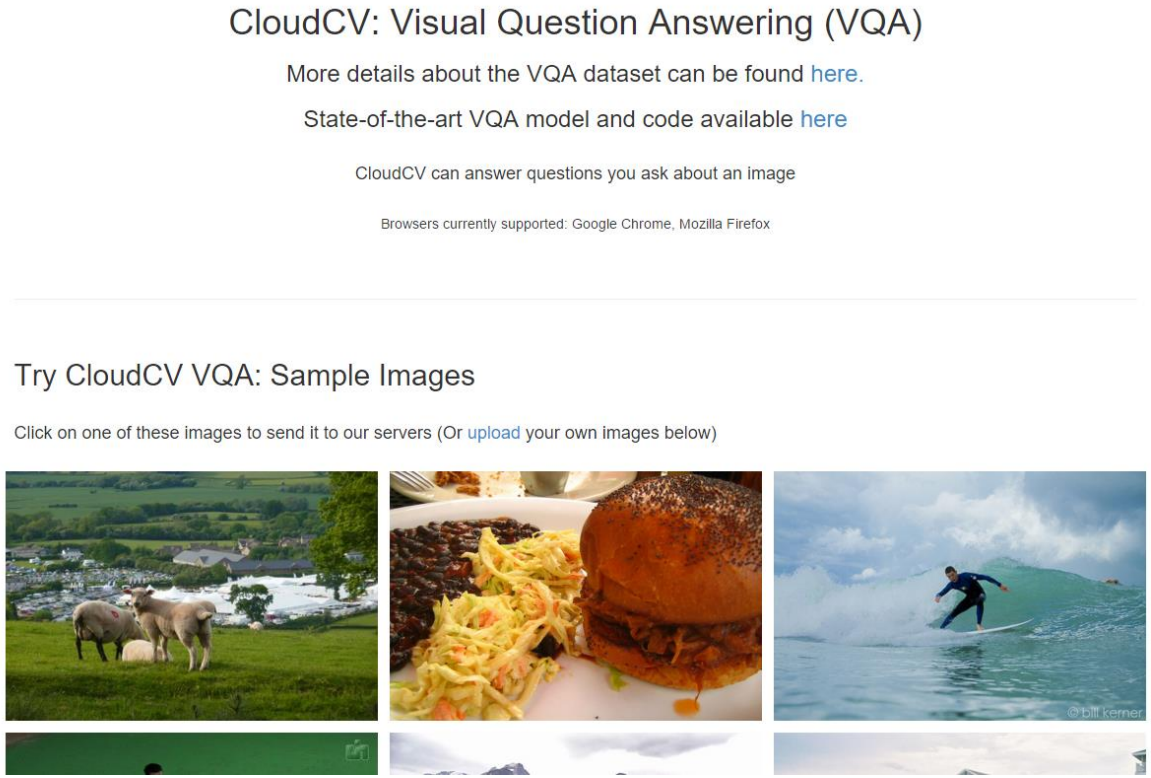
Microsoft Deep Learning Semantic Image Segmentation

Image captioning and Q&A

Click to go to the video in Youtube

NeuralTalk and Walk, recognition, text description of the image while walking

Click to go to the website



Why should we be impressed?

- Vision is ultra challenging!
  - For a small 256x256 resolution and for 256 pixel values
  - a total $2^{524,288}$ of possible images
  - In comparison there are about 10^{24}stars in the universe
- Visual object variations
  - Different viewpoints, scales, deformations, occlusions
- Semantic object variations
  - Intra-class variation
  - Inter-class overlaps

Self-driving cars

Click to go to the video in Youtube

Self Driving Cars HD

Drones and robots

Click to go to the video in Youtube

Deep Sensorimotor Learning

Game AI

Click to go to the video in Youtube

Google DeepMind's Deep Q-learning playing Atari Breakout

Why should we be impressed?

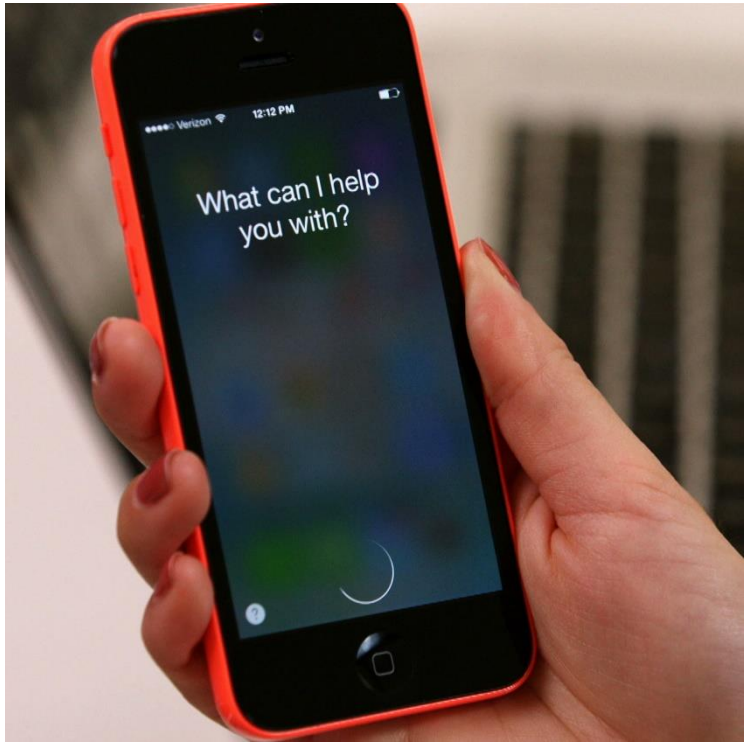
- Typically robotics are considered in controlled environments
  - Laboratory settings, Predictable positions, Standardized tasks (like in factory robots)
- What about real life situations?
  - Environments constantly change, new tasks need to be learnt without guidance, unexpected factors must be dealt with
- Game AI
  - At least $10^{10^{48}}$ possible GO games. Where do we even start?

Word and sentence representations

Newspapers			
New York	New York Times	Baltimore	Baltimore Sun
San Jose	San Jose Mercury News	Cincinnati	Cincinnati Enquirer
NHL Teams			
Boston	Boston Bruins	Montreal	Montreal Canadiens
Phoenix	Phoenix Coyotes	Nashville	Nashville Predators
NBA Teams			
Detroit	Detroit Pistons	Toronto	Toronto Raptors
Oakland	Golden State Warriors	Memphis	Memphis Grizzlies
Airlines			
Austria	Austrian Airlines	Spain	Spainair
Belgium	Brussels Airlines	Greece	Aegean Airlines
Company executives			
Steve Ballmer	Microsoft	Larry Page	Google
Samuel J. Palmisano	IBM	Werner Vogels	Amazon

Table 2: Examples of the analogical reasoning task for phrases (the full test set has 3218 examples). The goal is to compute the fourth phrase using the first three. Our best model achieved an accuracy of 72% on this dataset.

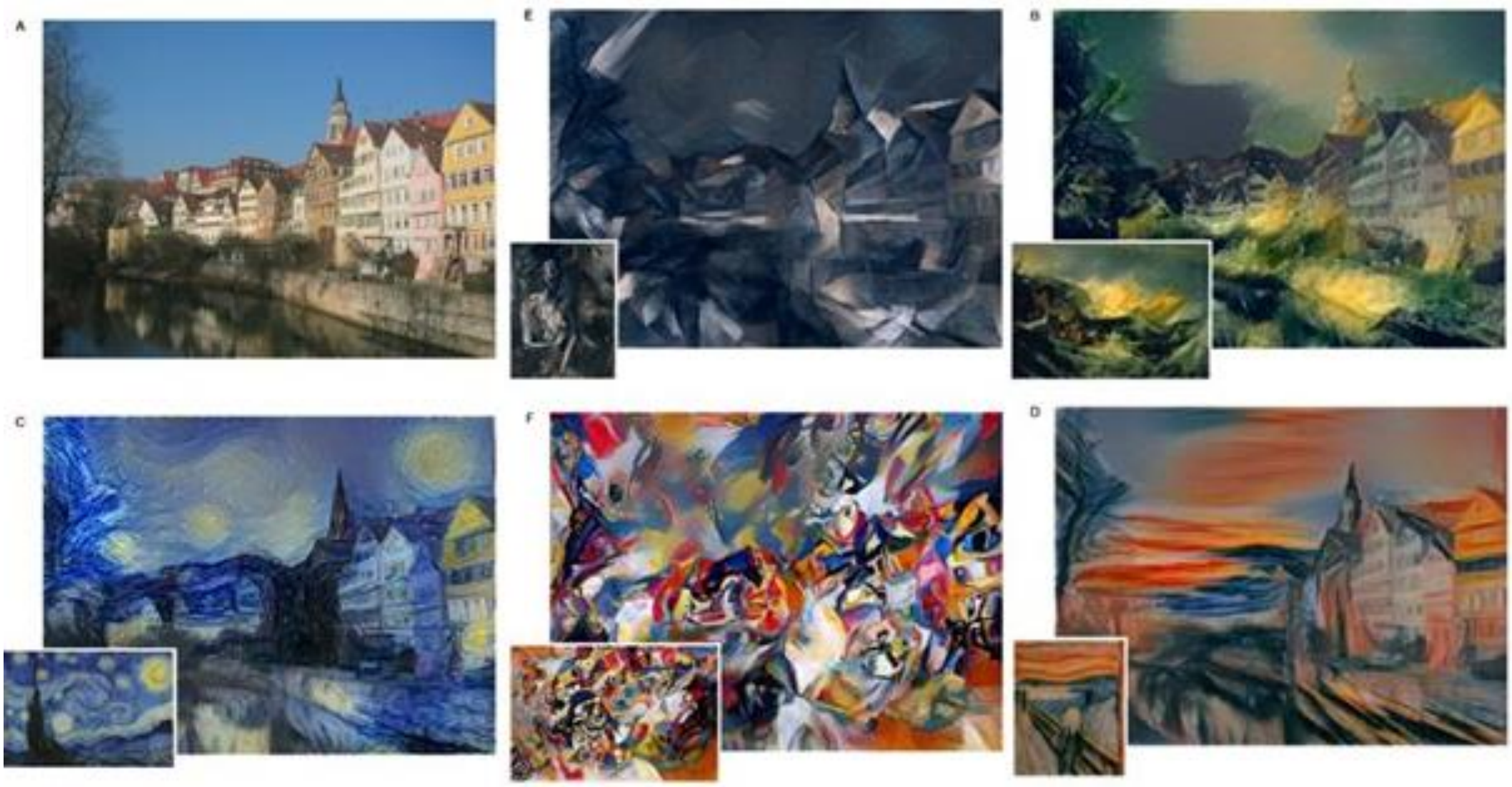
Speech recognition and Machine translation



Why should we be impressed?

- NLP is an extremely complex task
  - synonymy (“chair”, “stool” or “beautiful”, “handsome”)
  - ambiguity (“I made her duck”, “Cut to the chase”)
- NLP is very high dimensional
  - assuming 150K english words, we need to learn 150K classifiers
  - with quite sparse data for most of them
- Beating NLP feels the closest to achieving true AI
  - although true AI probably goes beyond NLP, Vision, Robotics, … alone

Imitating famous painters



https://www.youtube.com/watch?v=-R9bJGNHltQ

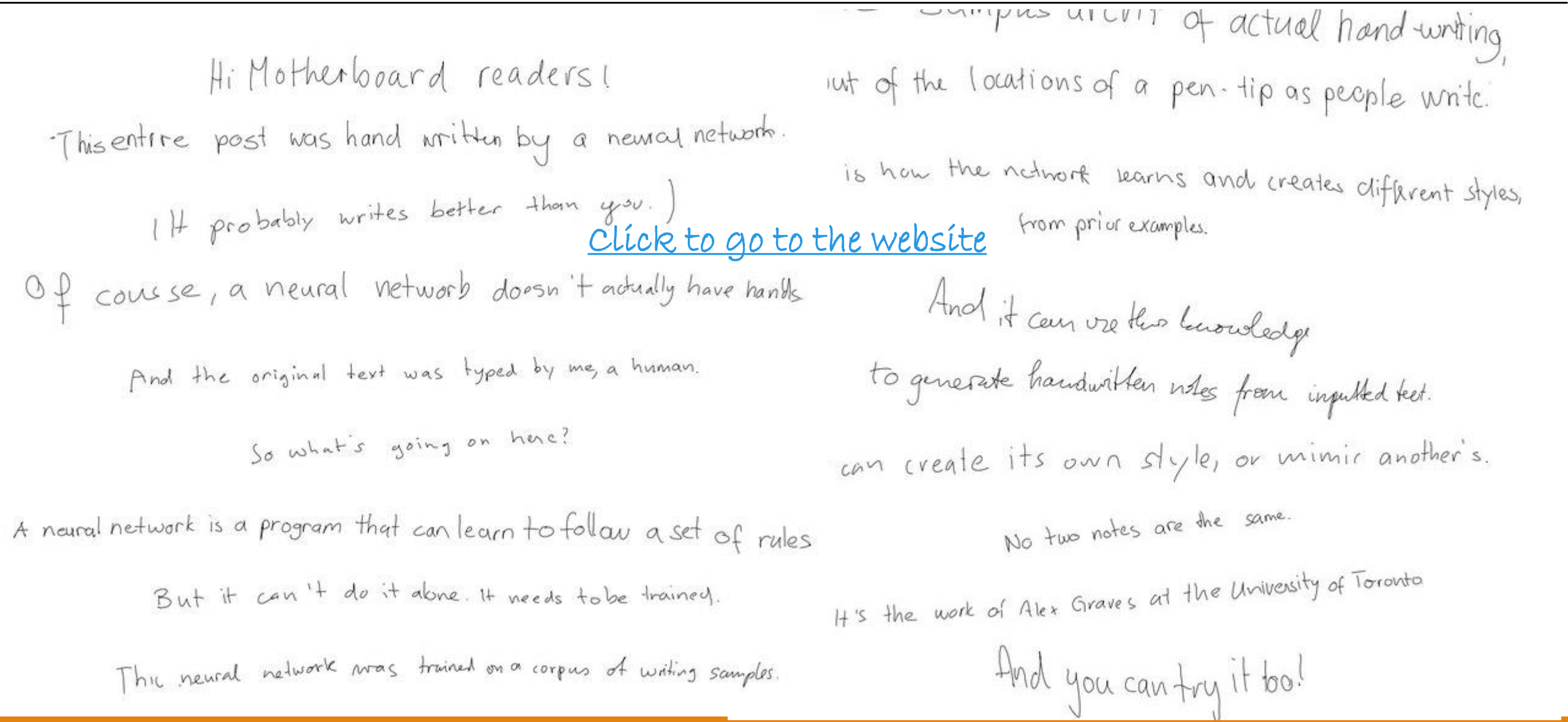
Or dreaming …

Click to go to the video in Youtube

Journey Through the Layers of the Mind

https://www.youtube.com/watch?v=2VDoi96j6rQ

Handwriting



Why should we be impressed?

- Music, painting, etc. are tasks that are uniquely human
  - Difficult to model
  - Even more difficult to evaluate (if not impossible)
- If machines can generate novel pieces even remotely resembling art, they must have understood something about “beauty”, “harmony”, etc.
- Have they really learned to generate new art, however?
  - Or do they just fool us with their tricks?

What is Deep Learning?

Part of the machine learning field of learning representations of data. Exceptional effective at learning patterns.

Utilizes learning algorithms that derive meaning out of data by using a hierarchy of multiple layers that mimic the neural networks of our brain.

If you provide the system tons of information, it begins to understand it and respond in useful ways.

The Big Players

Superstar Researchers

Geoffrey Hinton: University of Toronto & Google

Yann LeCun: New York University & Facebook

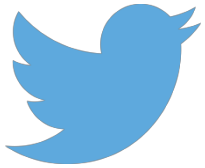
Andrew Ng: Stanford & Baidu

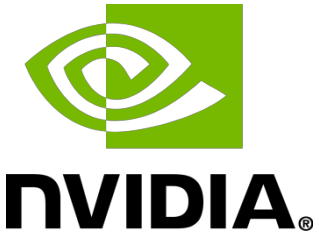
Yoshua Bengio: University of Montreal

Jürgen Schmidhuber: Swiss AI Lab & NNAISENSE

The Big Players

Companies



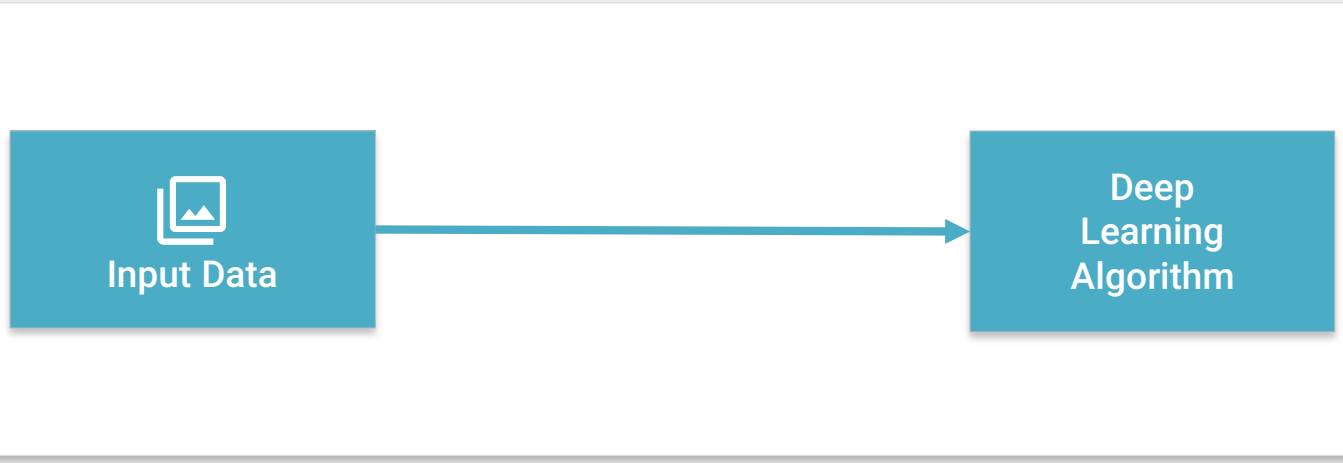


The Big Players

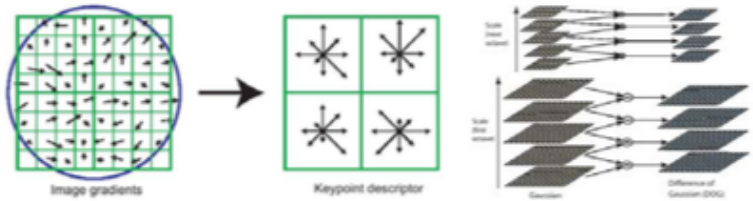
Startups

Deep Learning - Basics

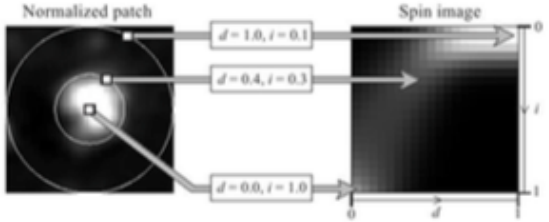
No more feature engineering



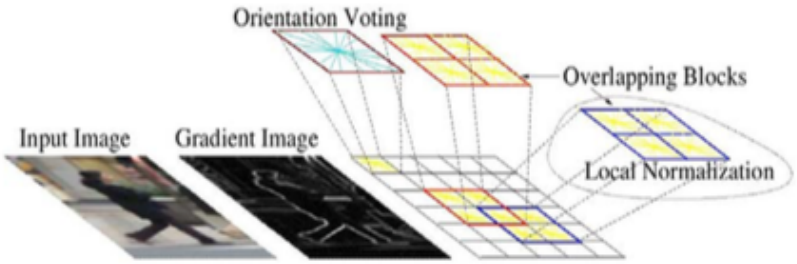
Feature Representations



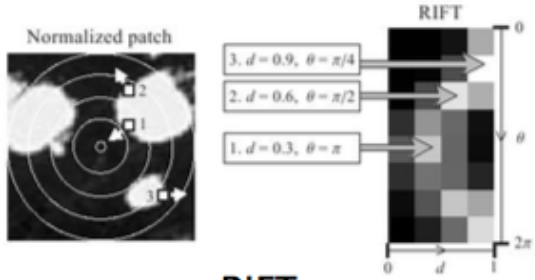
SIFT



Spin image

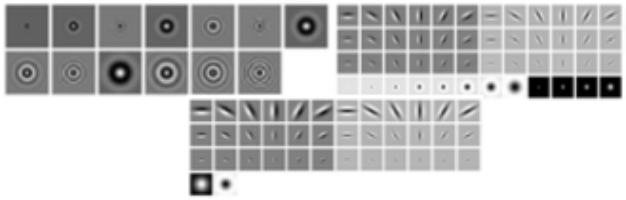


HoG

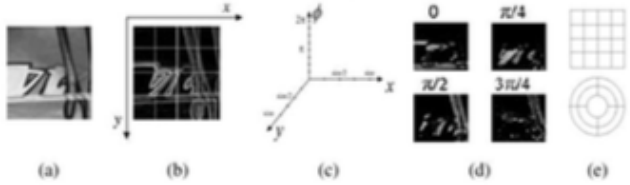


RIFT

Expert Knowledge!



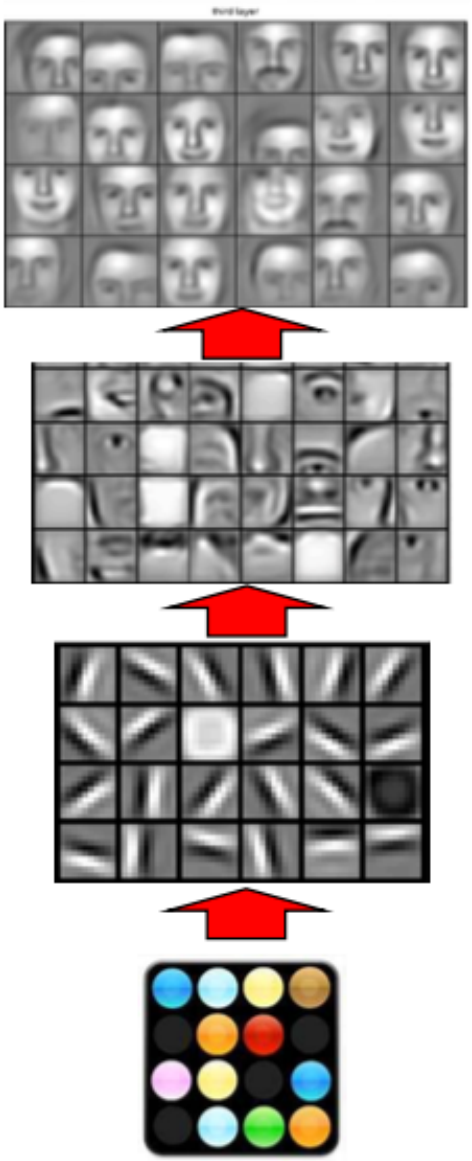
Textons



GLOH

Source: feature representations in computer vision(Honglak lee)

Deep Learning: learn representations!



Source:Lee et.al. ICML2009

A bit of history:

Hubel & Wiesel,

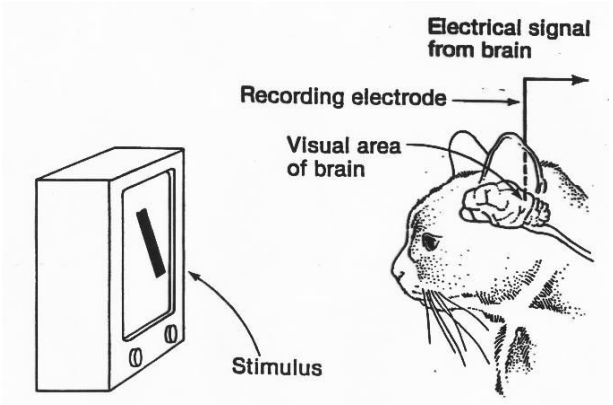
1959

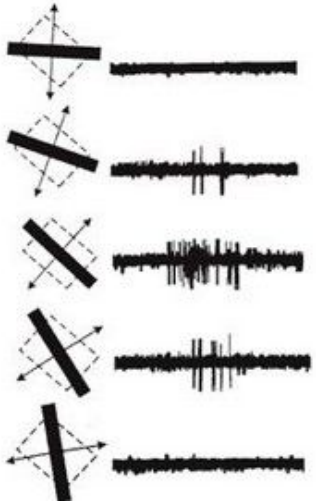
RECEPTIVE FIELDS OF SINGLE NEURONES IN
THE CAT'S STRIATE CORTEX

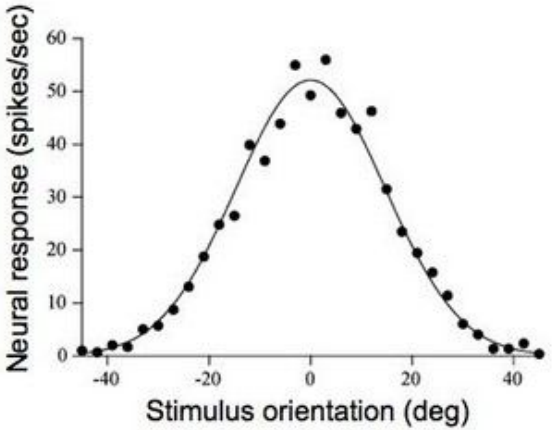
1962

RECEPTIVE FIELDS, BINOCULAR INTERACTION
AND FUNCTIONAL ARCHITECTURE IN
THE CAT'S VISUAL CORTEX

1968...

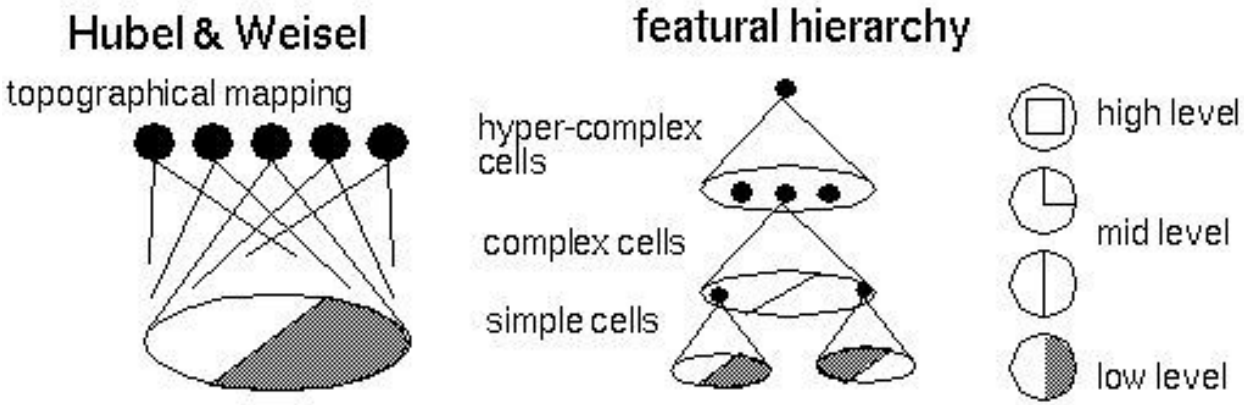






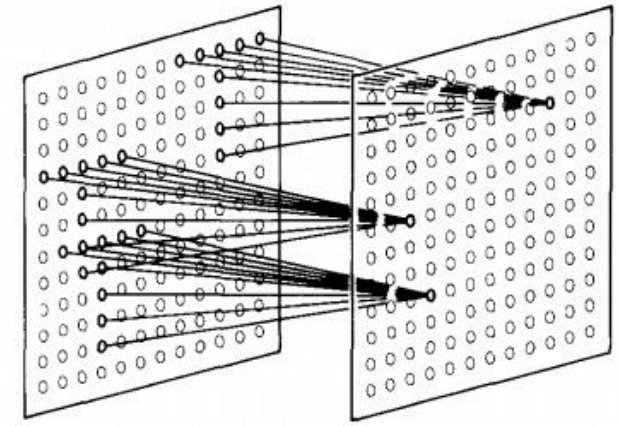
Video time https://youtu.be/8VdFf3egwfg?t=1m10s

Hierarchical organization

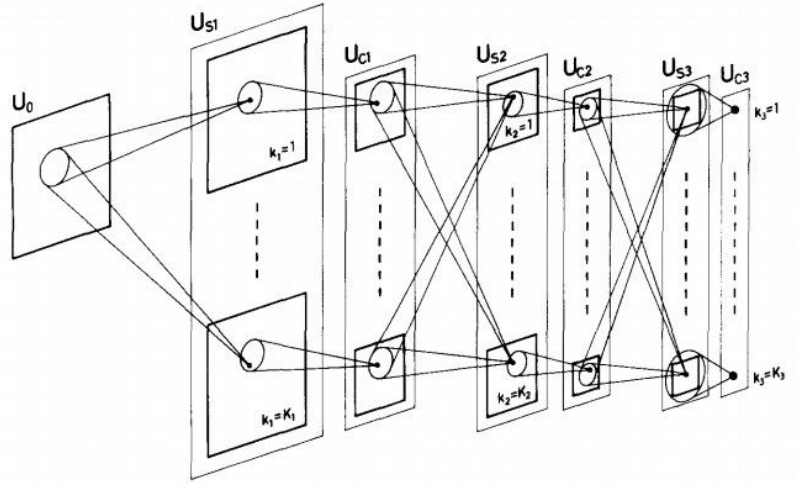


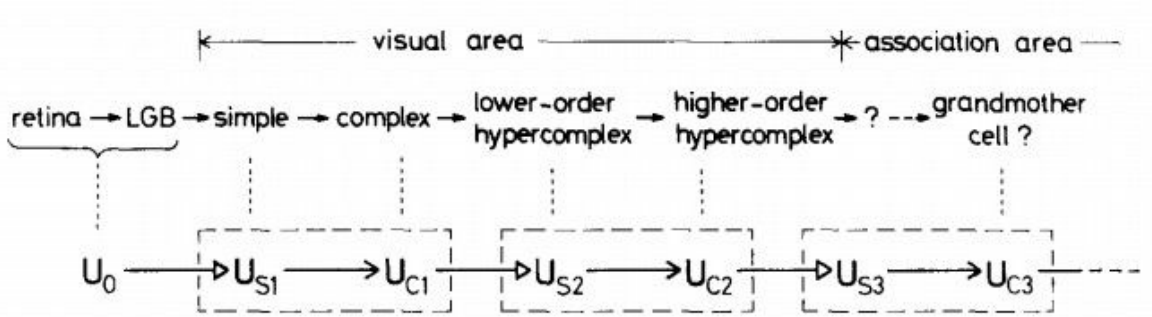
A bit of history:

Neurocognitron
[Fukushima 1980]



"sandwich" architecture (SCSCSC…)
simple cells: modifiable parameters
complex cells: perform pooling

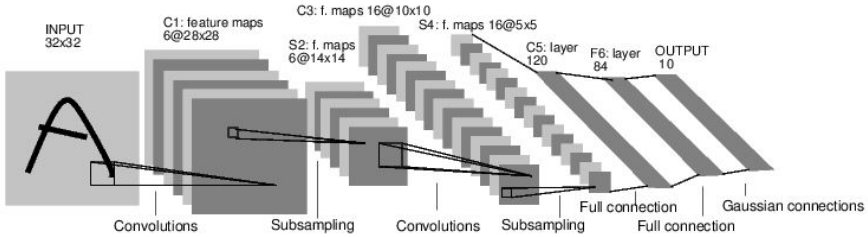




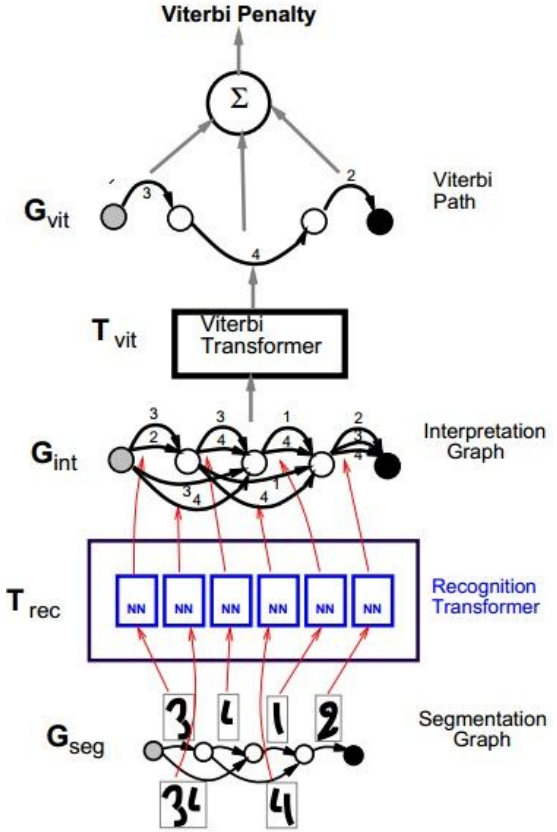
A bit of history:

Gradient-based learning applied to document recognition

[LeCun, Bottou, Bengio, Haffner 1998]



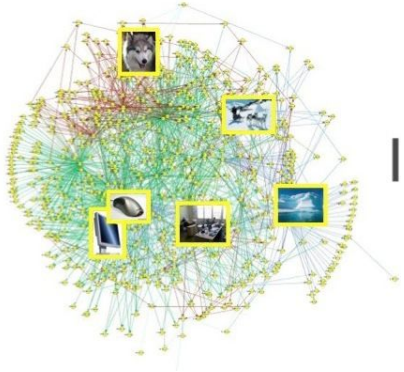
LeNet-5



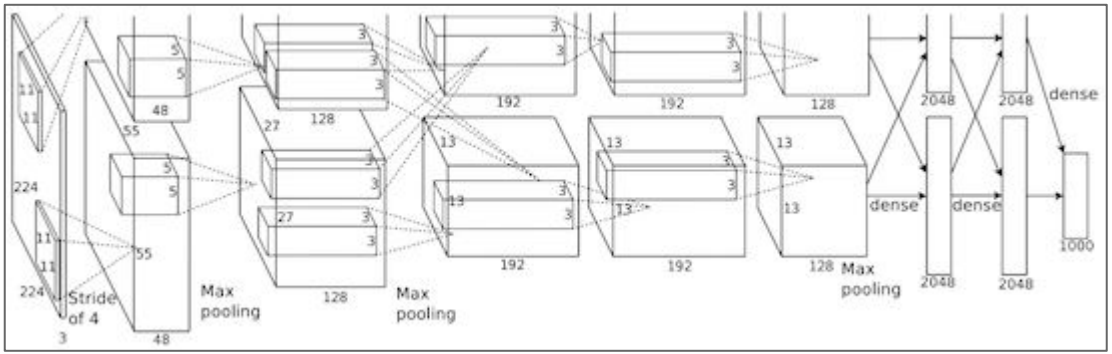
A bit of history:

ImageNet Classification with Deep Convolutional Neural Networks

[Krizhevsky, Sutskever, Hinton, 2012]



IMAGENET



"AlexNet"

CONVOLUTIONAL NEURAL NETWORK

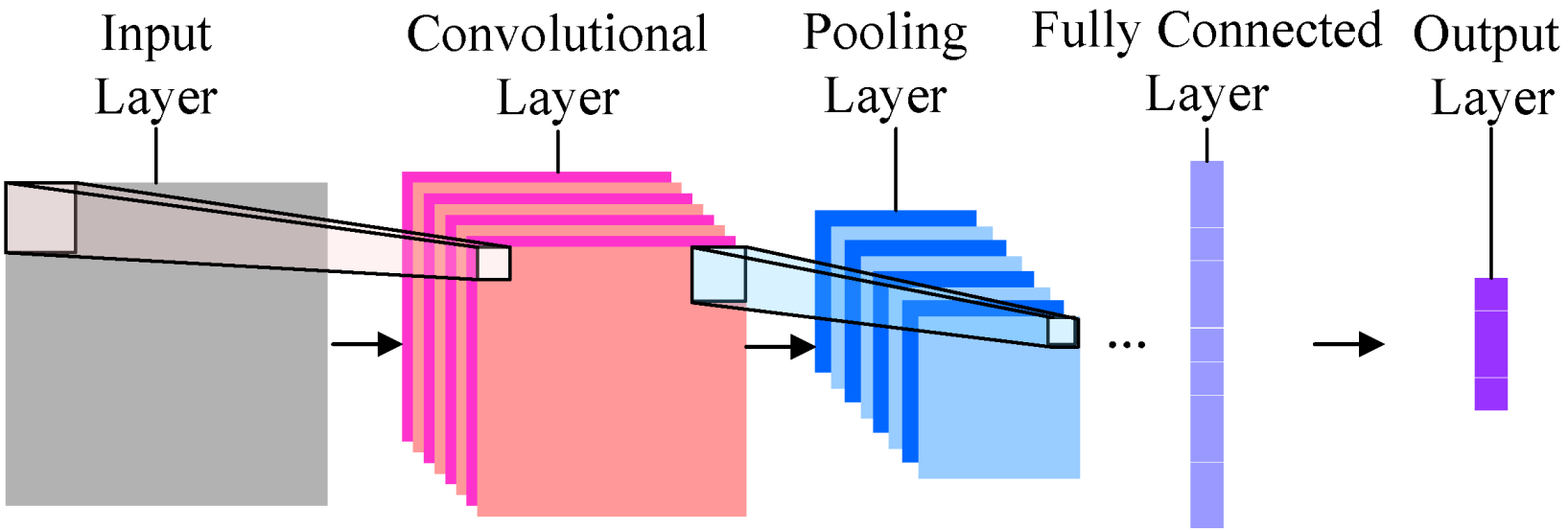
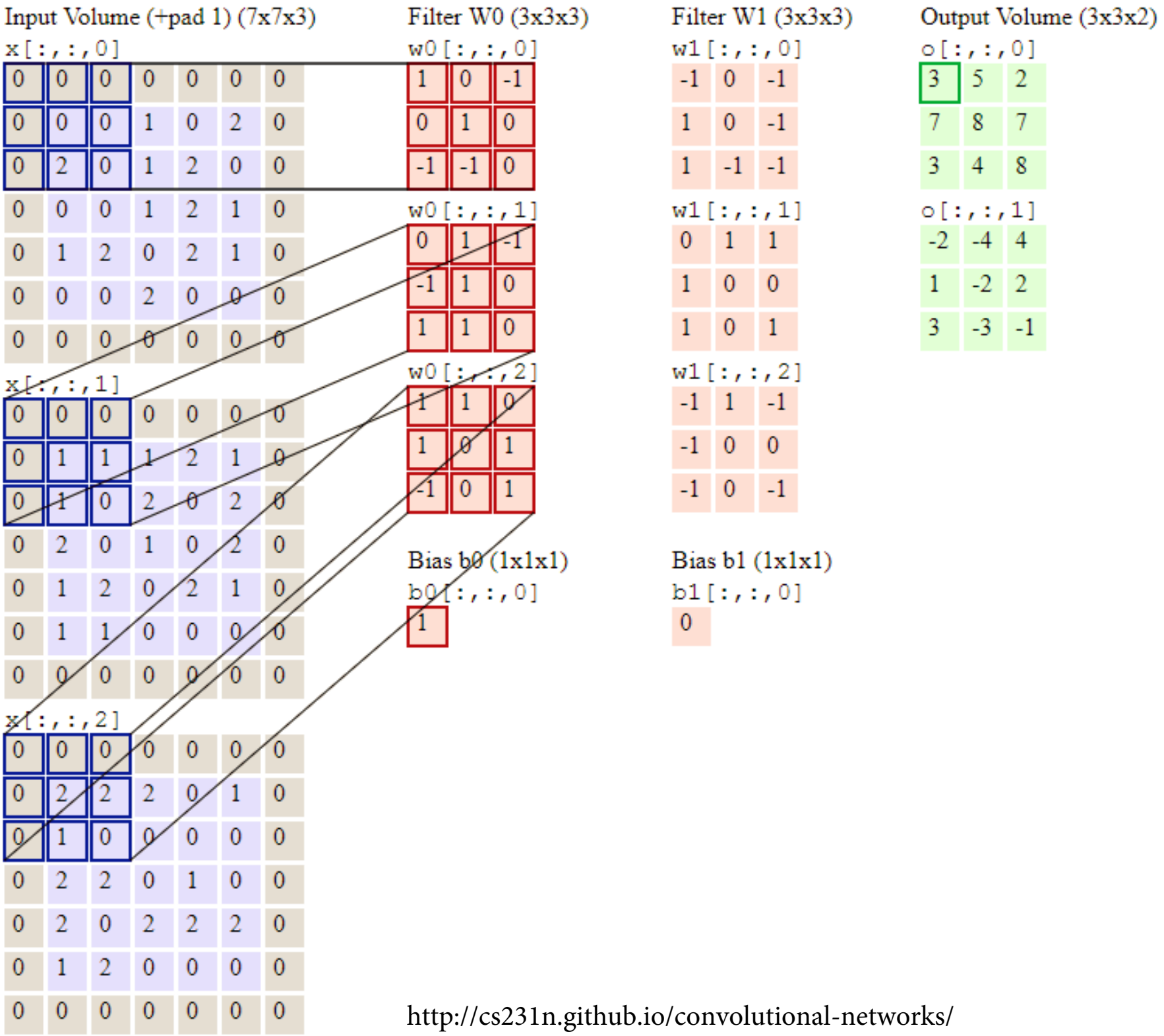
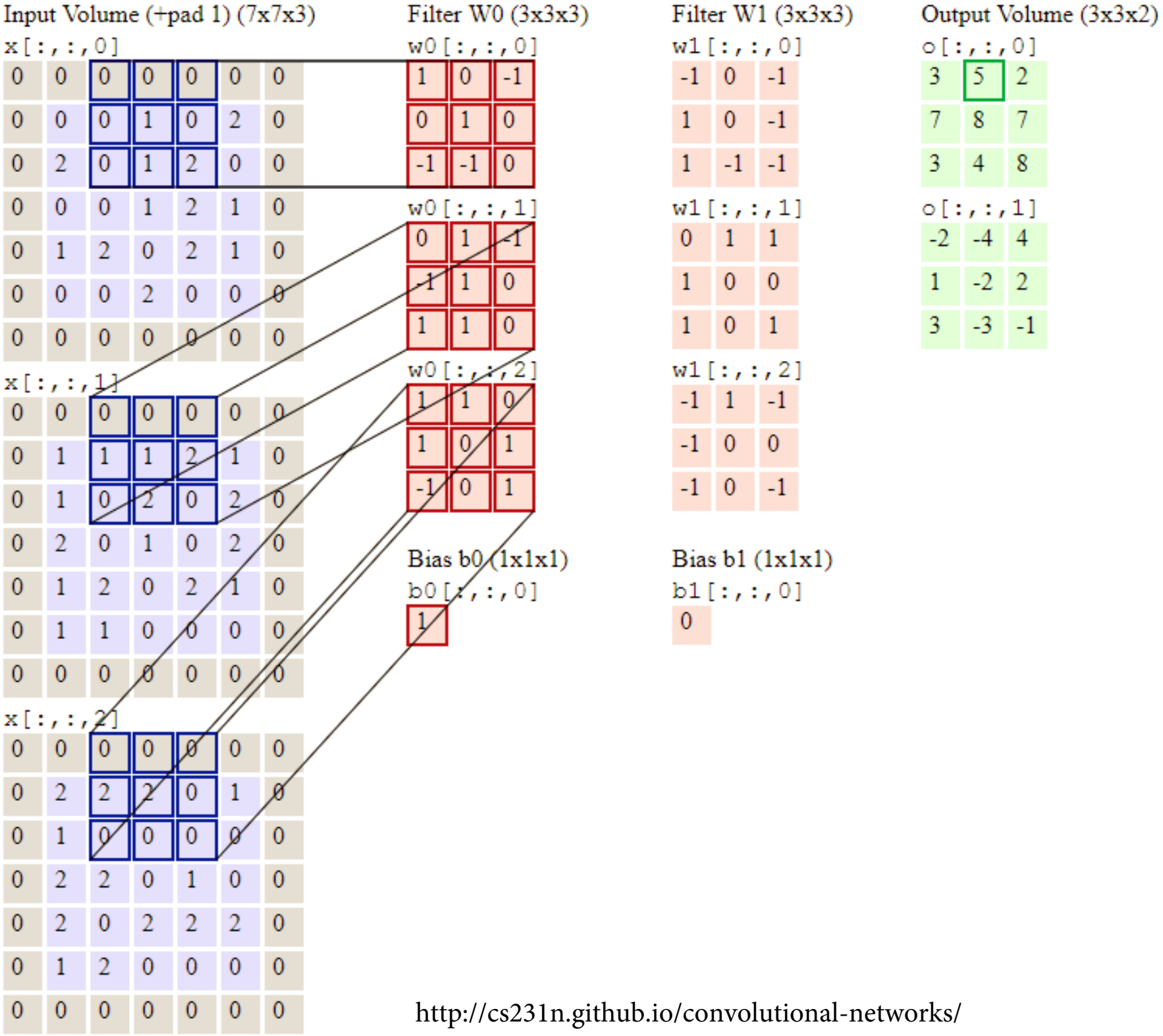


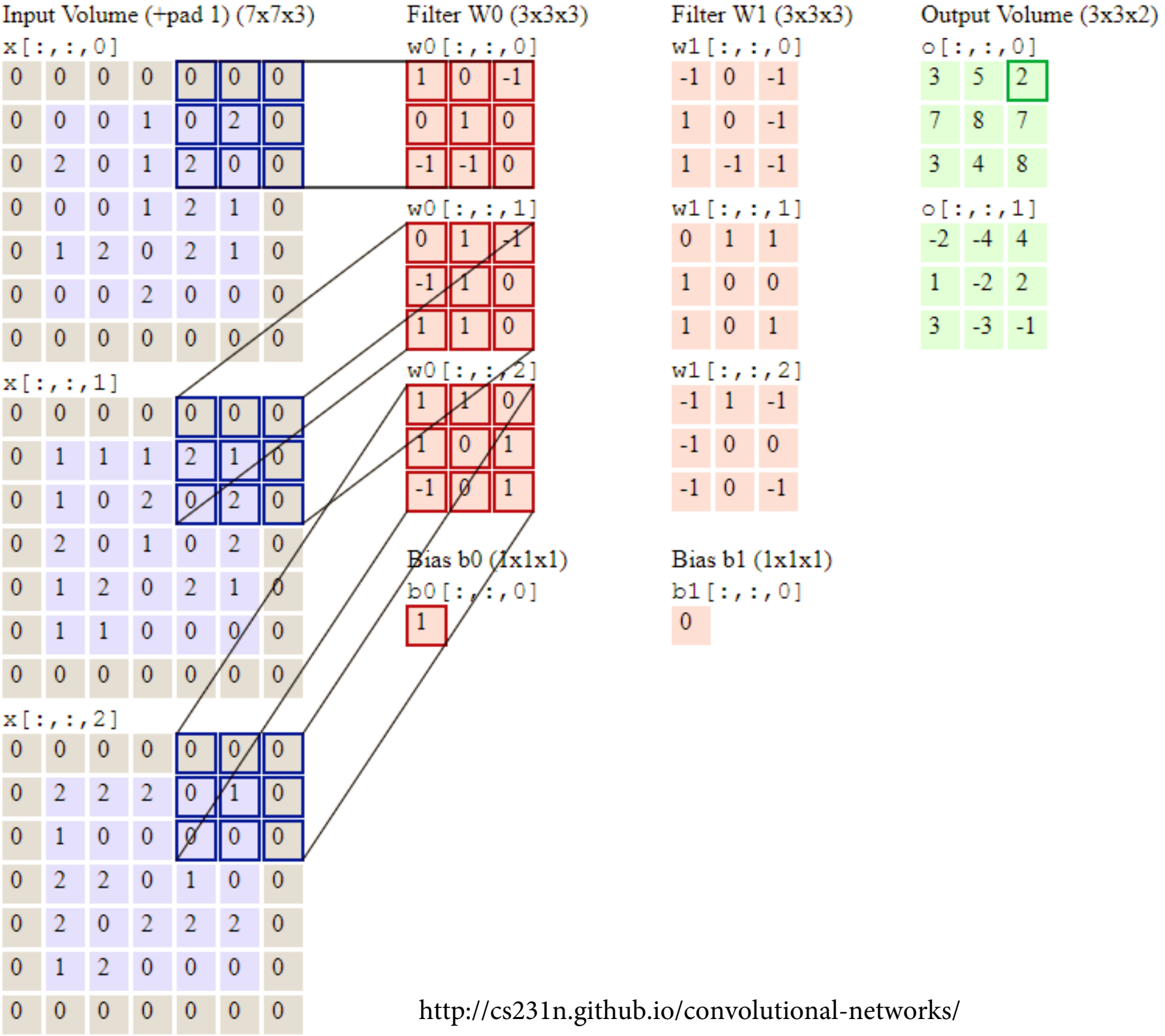
Imagem: http://www.mdpi.com/2078-2489/7/4/61



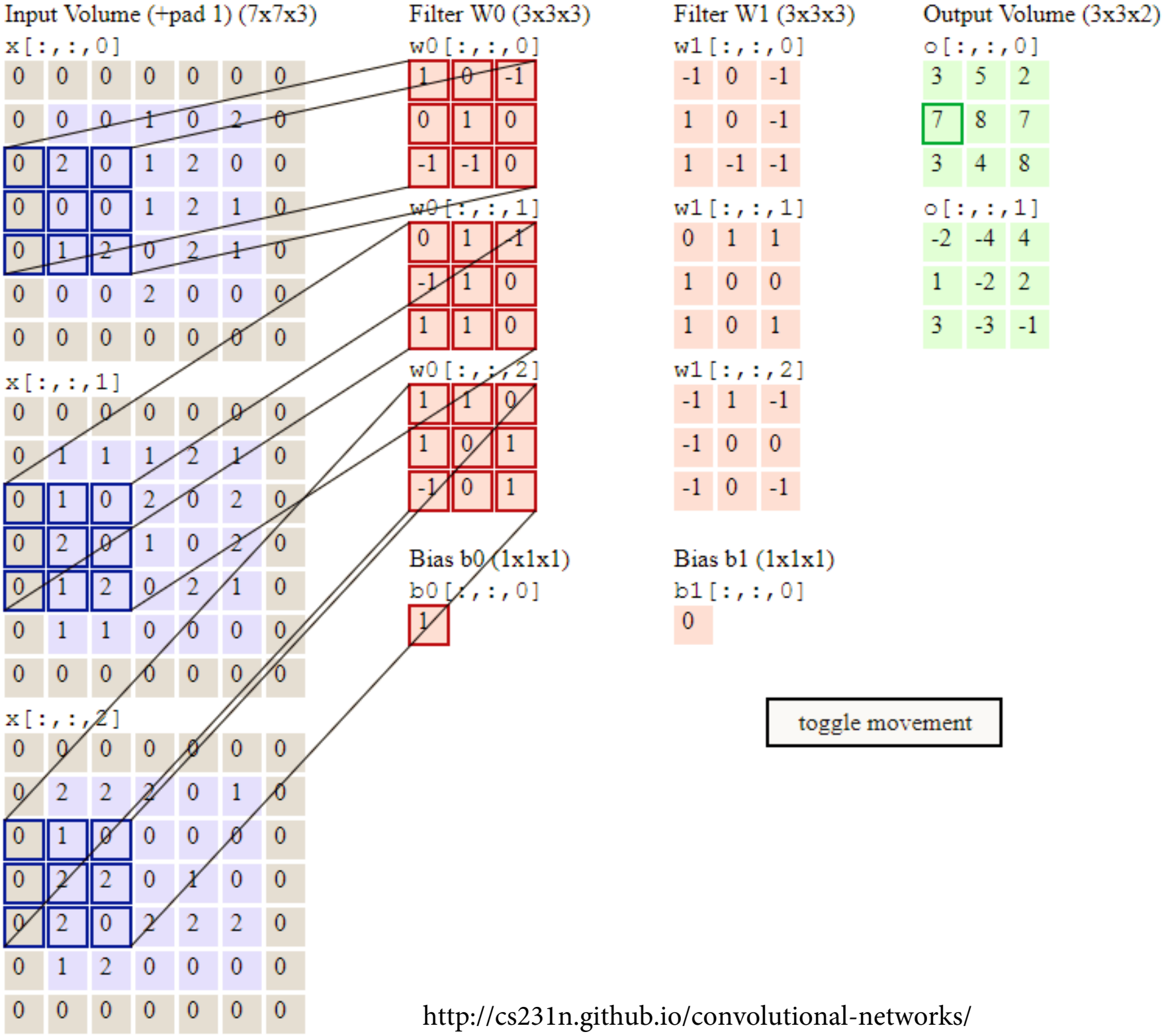
http://cs231n.github.io/convolutional-networks/



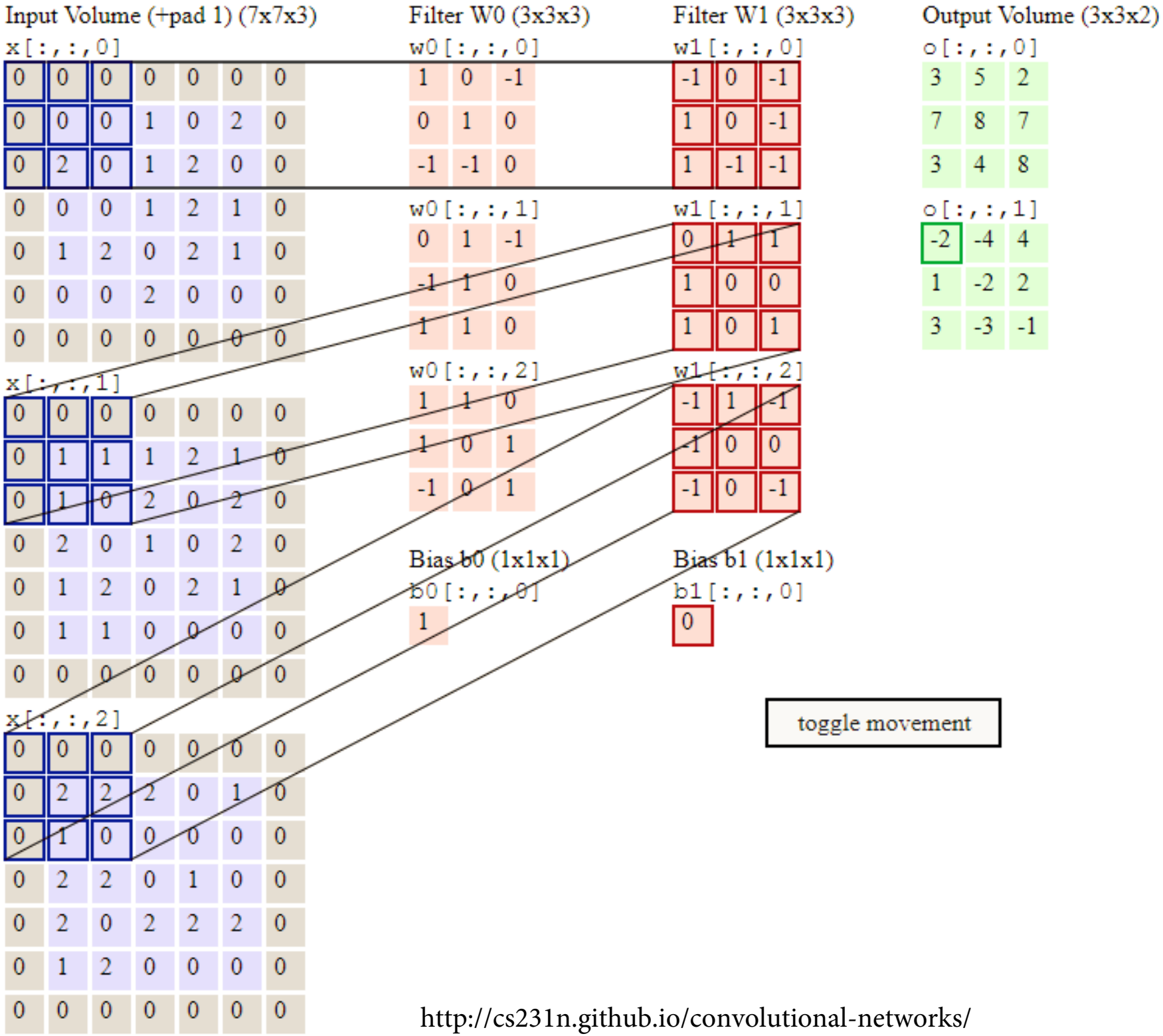
http://cs231n.github.io/convolutional-networks/



http://cs231n.github.io/convolutional-networks/

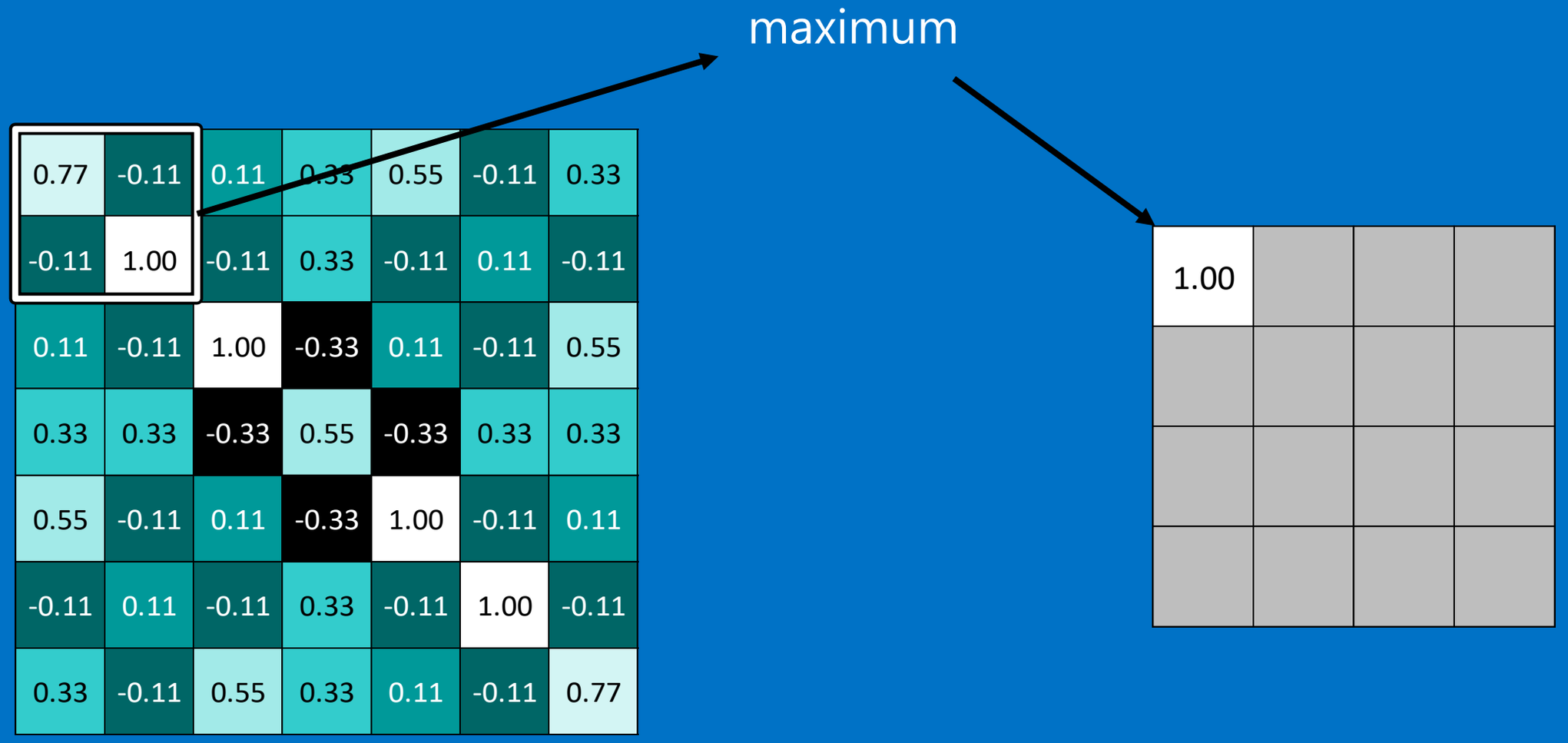


http://cs231n.github.io/convolutional-networks/



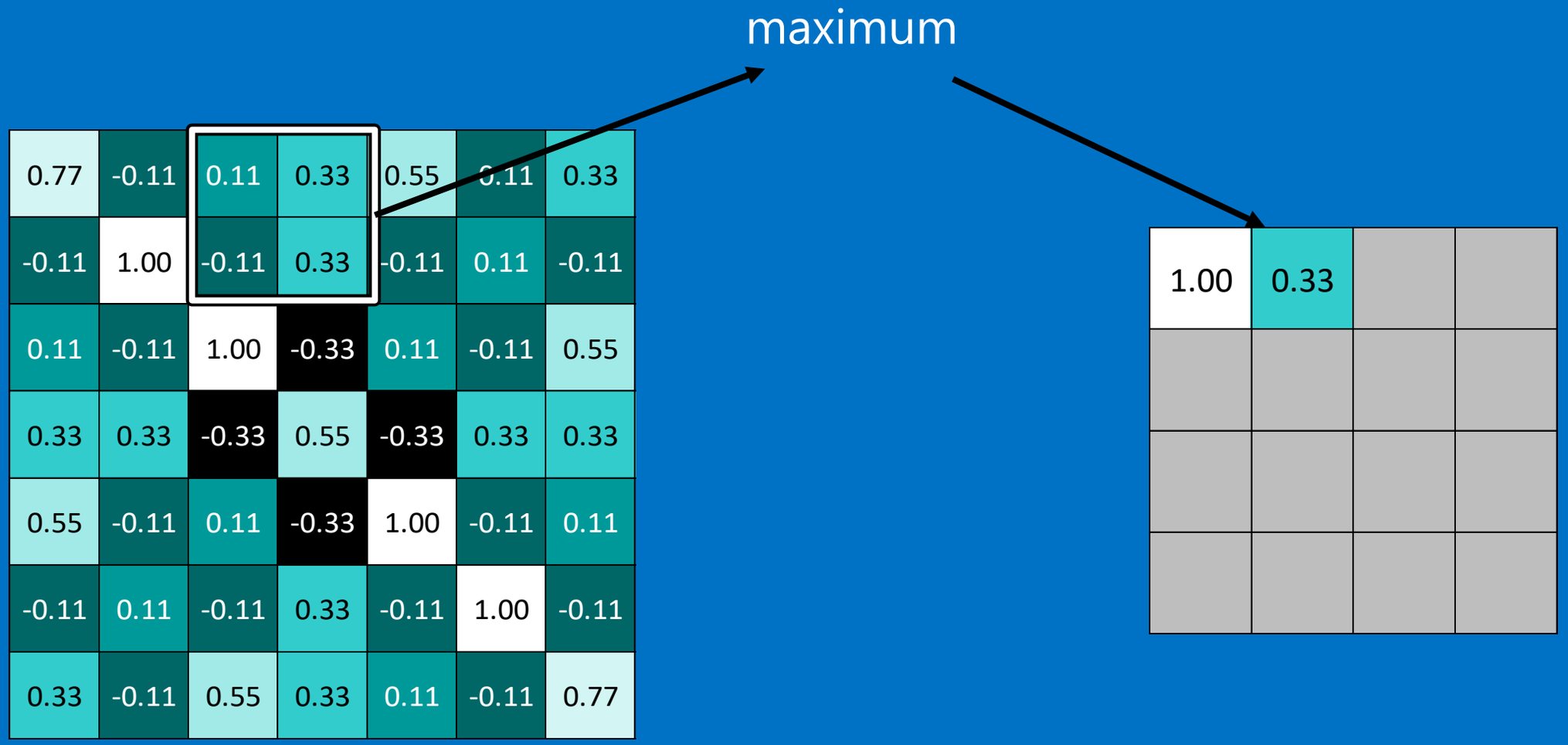
http://cs231n.github.io/convolutional-networks/

Pooling

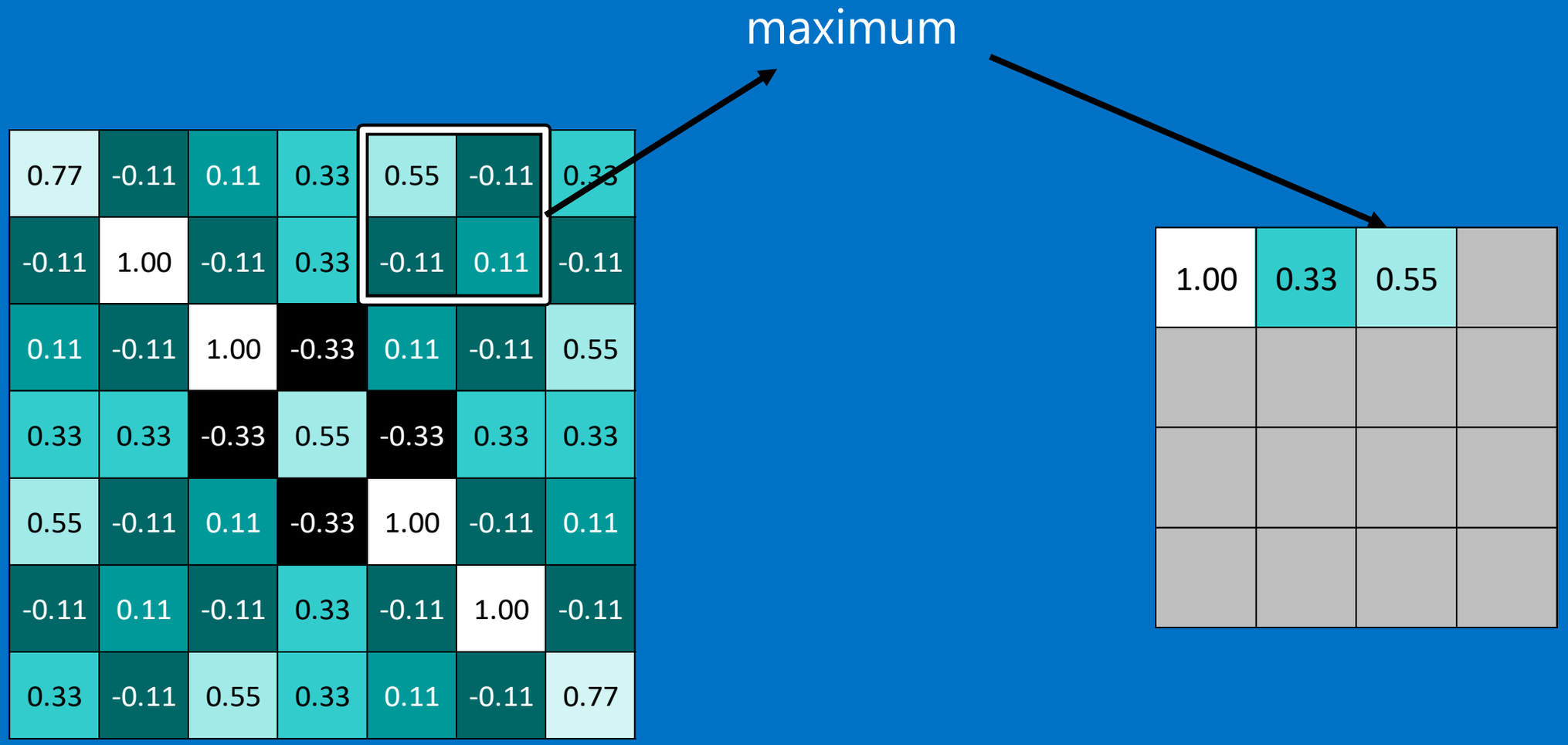


Brandon Rohrer

Pooling

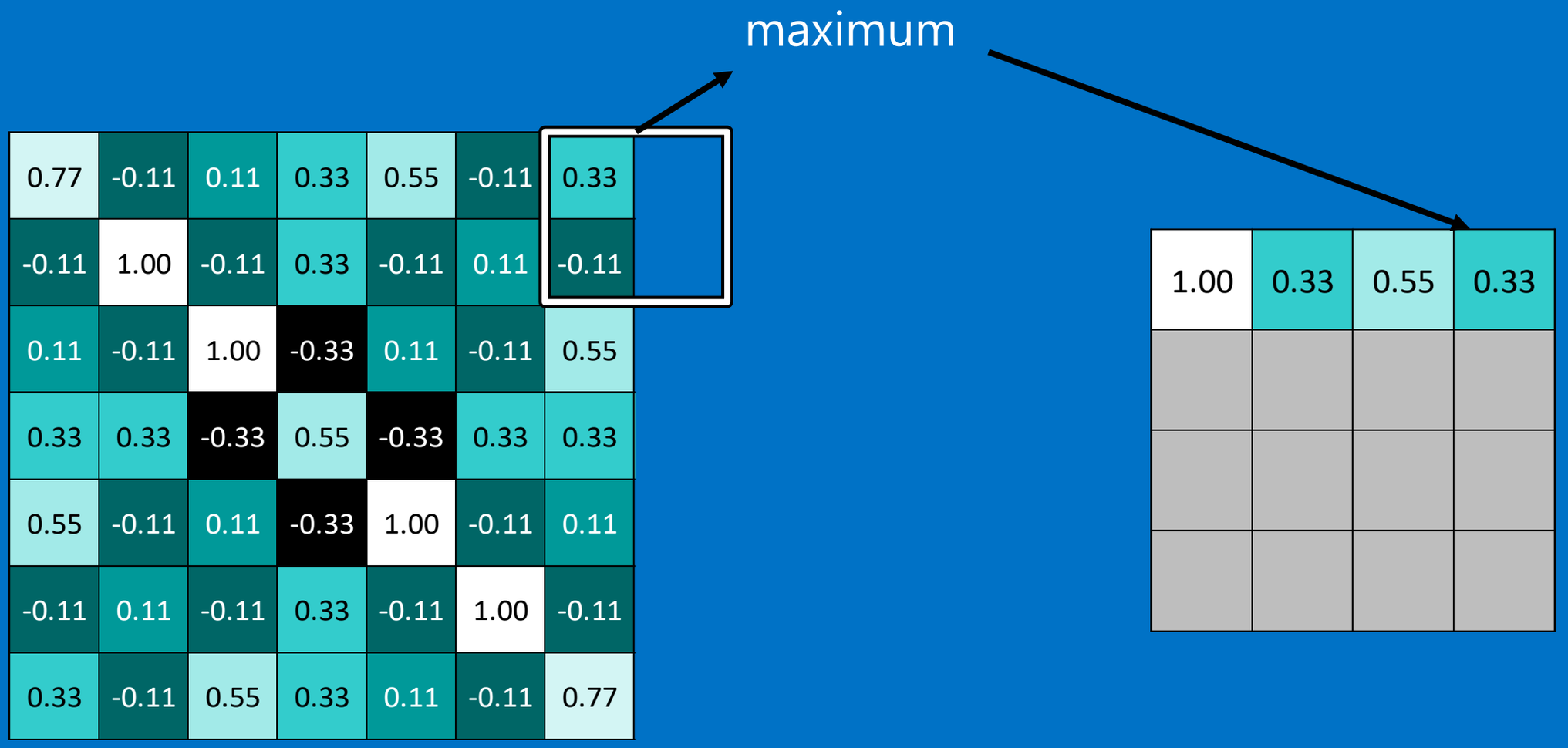

maximum
0.77 -0.11 0.11 0.33 0.55 0.11 0.33
-0.11 1.00 -0.11 0.33 -0.11 0.11 -0.11
0.11 -0.11 1.00 -0.33 0.11 -0.11 0.55
0.33 0.33 -0.33 0.55 -0.33 0.33 0.33
0.55 -0.11 0.11 -0.33 1.00 -0.11 0.11
-0.11 0.11 -0.11 0.33 -0.11 1.00 -0.11
0.33 -0.11 0.55 0.33 0.11 -0.11 0.77
1.00 0.33

Pooling

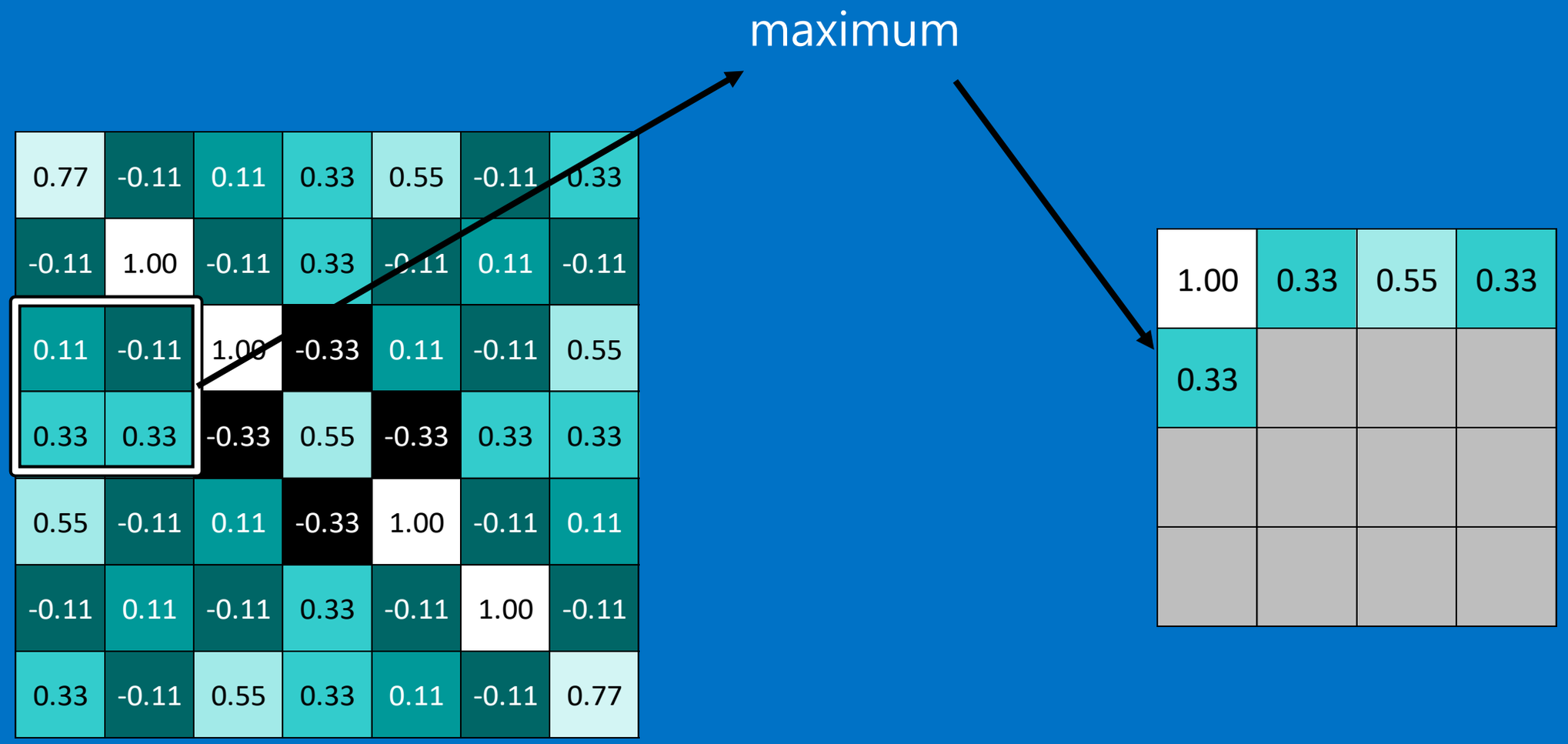


Brandon Rohrer

Pooling


maximum
0.77 -0.11 0.11 0.33 0.55 -0.11 0.33
-0.11 1.00 -0.11 0.33 -0.11 0.11 -0.11
0.11 -0.11 1.00 -0.33 0.11 -0.11 0.55
0.33 0.33 -0.33 0.55 -0.33 0.33 0.33
0.55 -0.11 0.11 -0.33 1.00 -0.11 0.11
-0.11 0.11 -0.11 0.33 -0.11 1.00 -0.11
0.33 -0.11 0.55 0.33 0.11 -0.11 0.77
1.00 0.33 0.55 0.33

Pooling

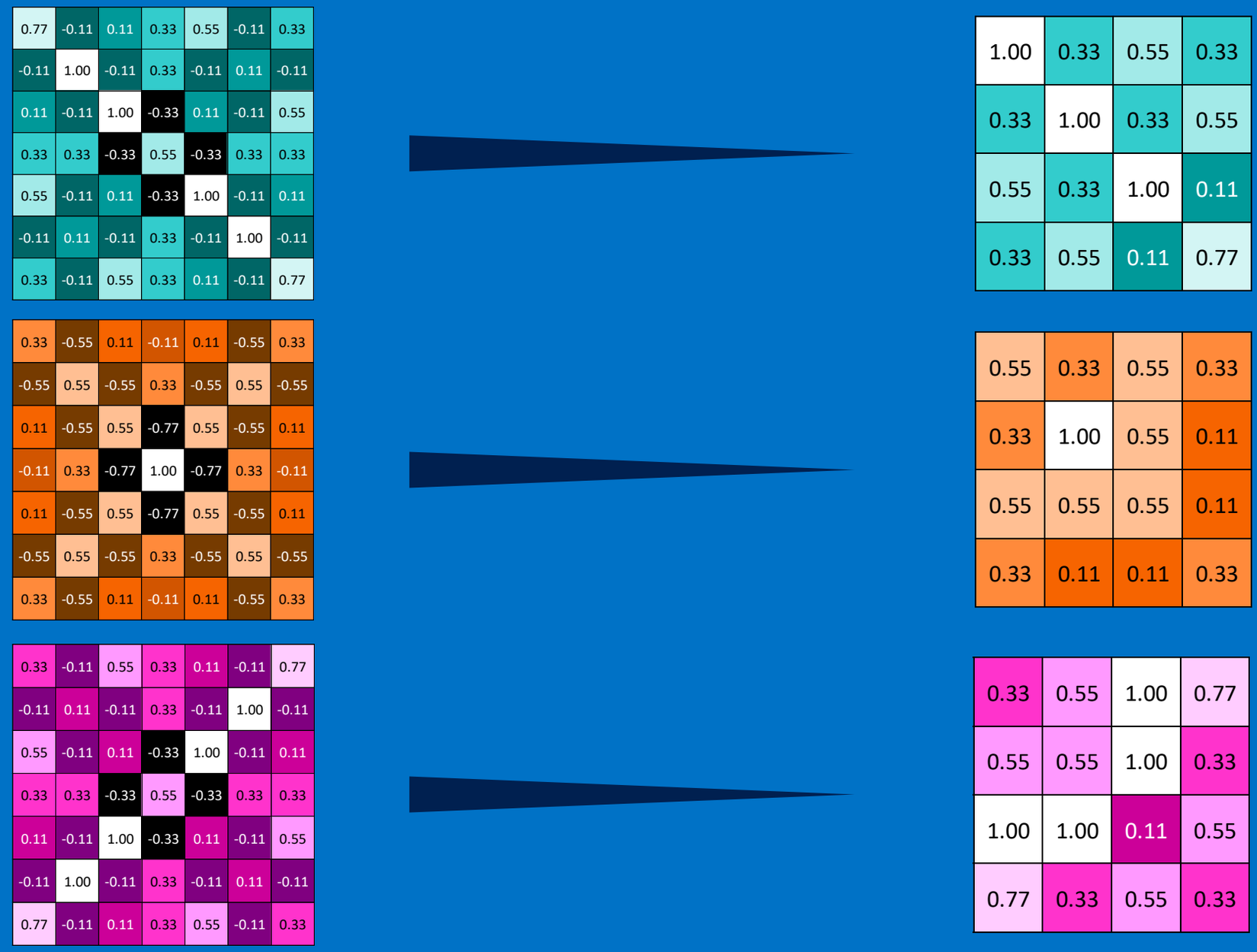


Brandon Rohrer

Pooling

0.77	-0.11	0.11	0.33	0.55	-0.11	0.33
-0.11	1.00	-0.11	0.33	-0.11	0.11	-0.11
0.11	-0.11	1.00	-0.33	0.11	-0.11	0.55
0.33	0.33	-0.33	0.55	-0.33	0.33	0.33
0.55	-0.11	0.11	-0.33	1.00	-0.11	0.11
-0.11	0.11	-0.11	0.33	-0.11	1.00	-0.11
0.33	-0.11	0.55	0.33	0.11	-0.11	0.77

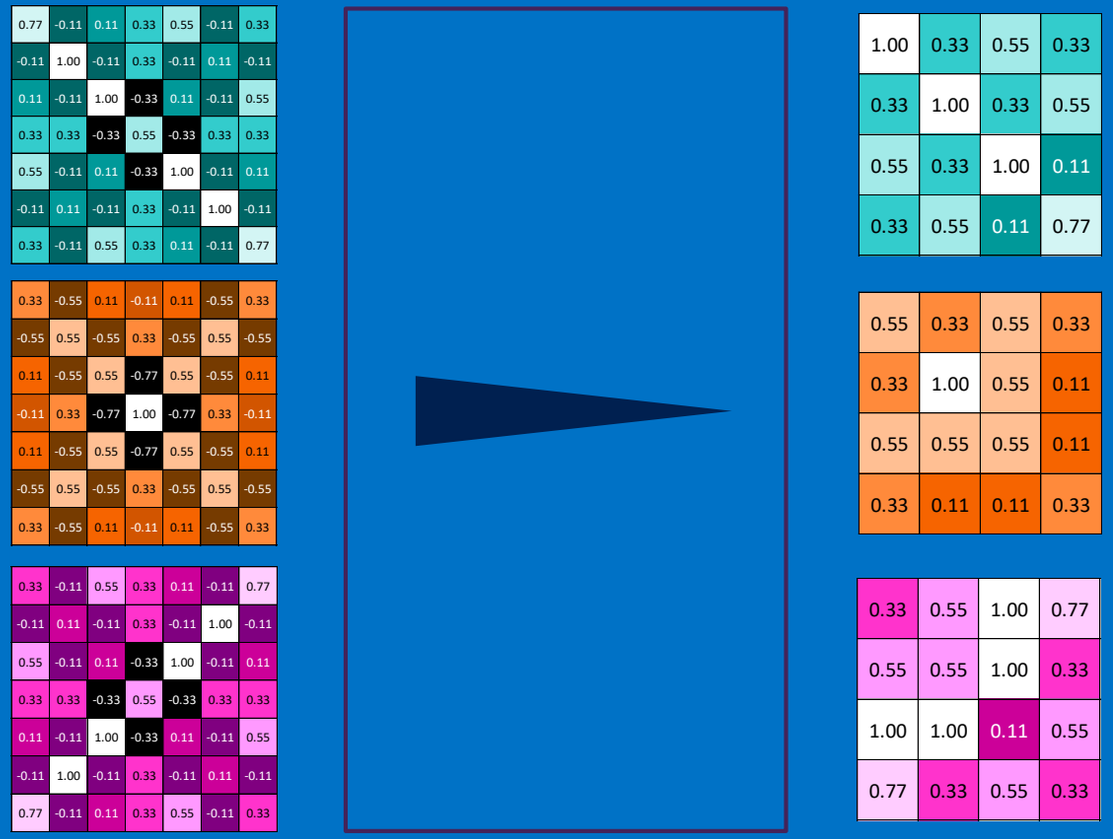
1.00	0.33	0.55	0.33
0.33	1.00	0.33	0.55
0.55	0.33	1.00	0.11
0.33	0.55	0.11	0.77



Brandon Rohrer

Pooling layer

A stack of images becomes a stack of smaller images.

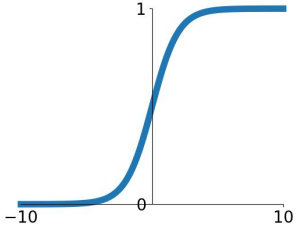


Brandon Rohrer

Activation Functions

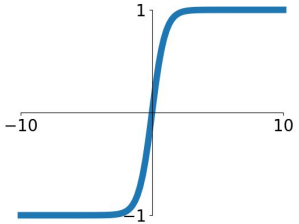
Sigmoid

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



tanh

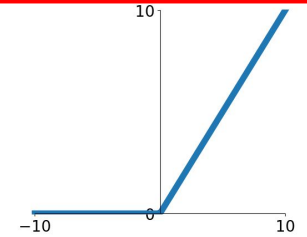
$$\tanh(x)$$



ReLU

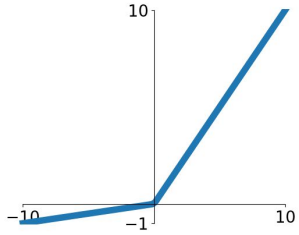
$$\max(0, x)$$

Good default choice



Leaky ReLU

$$\max(0.1x, x)$$

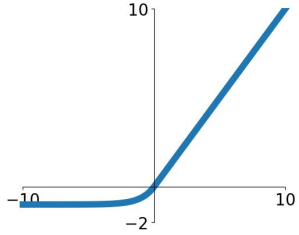


Maxout

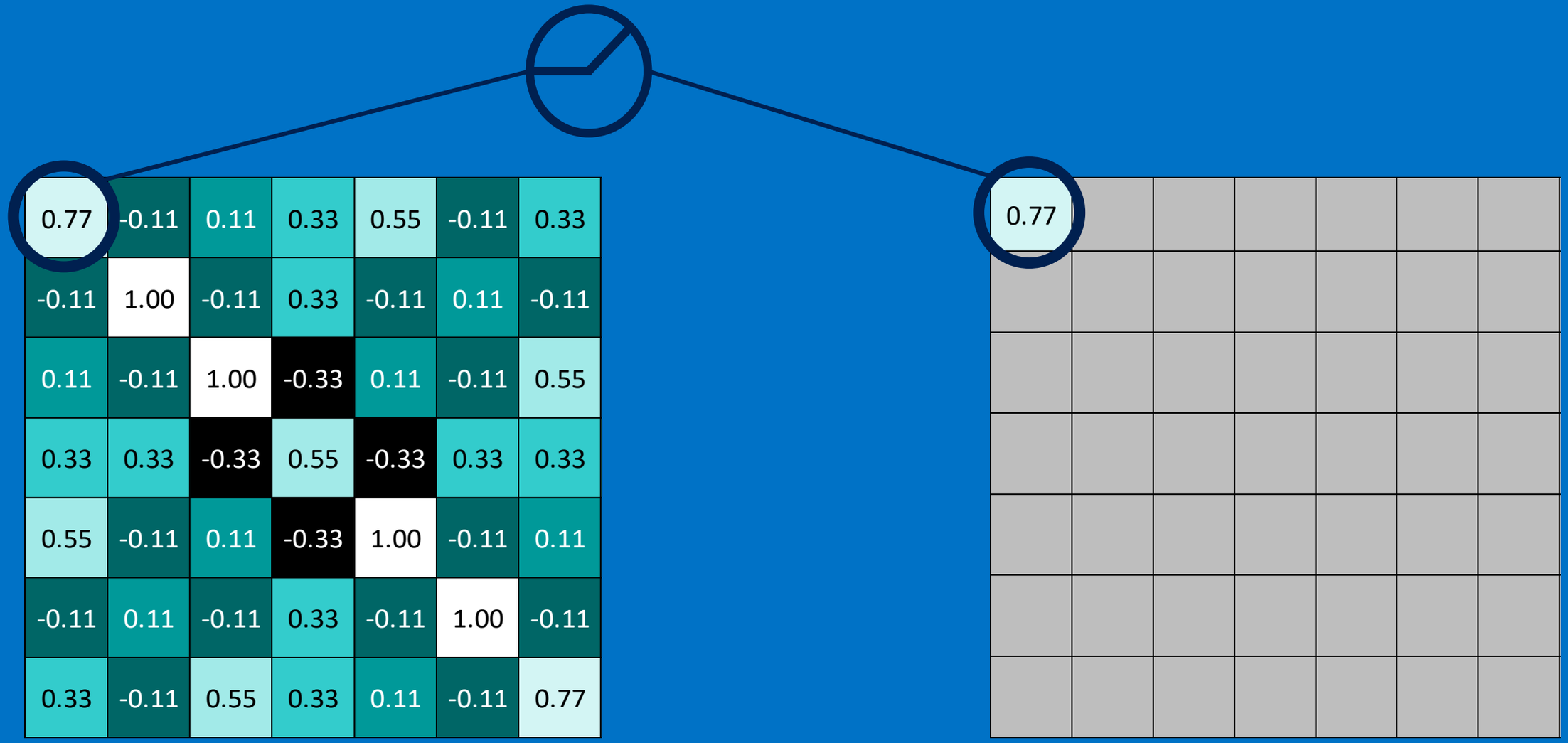
$$\max(w_1^T x + b_1, w_2^T x + b_2)$$

ELU

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

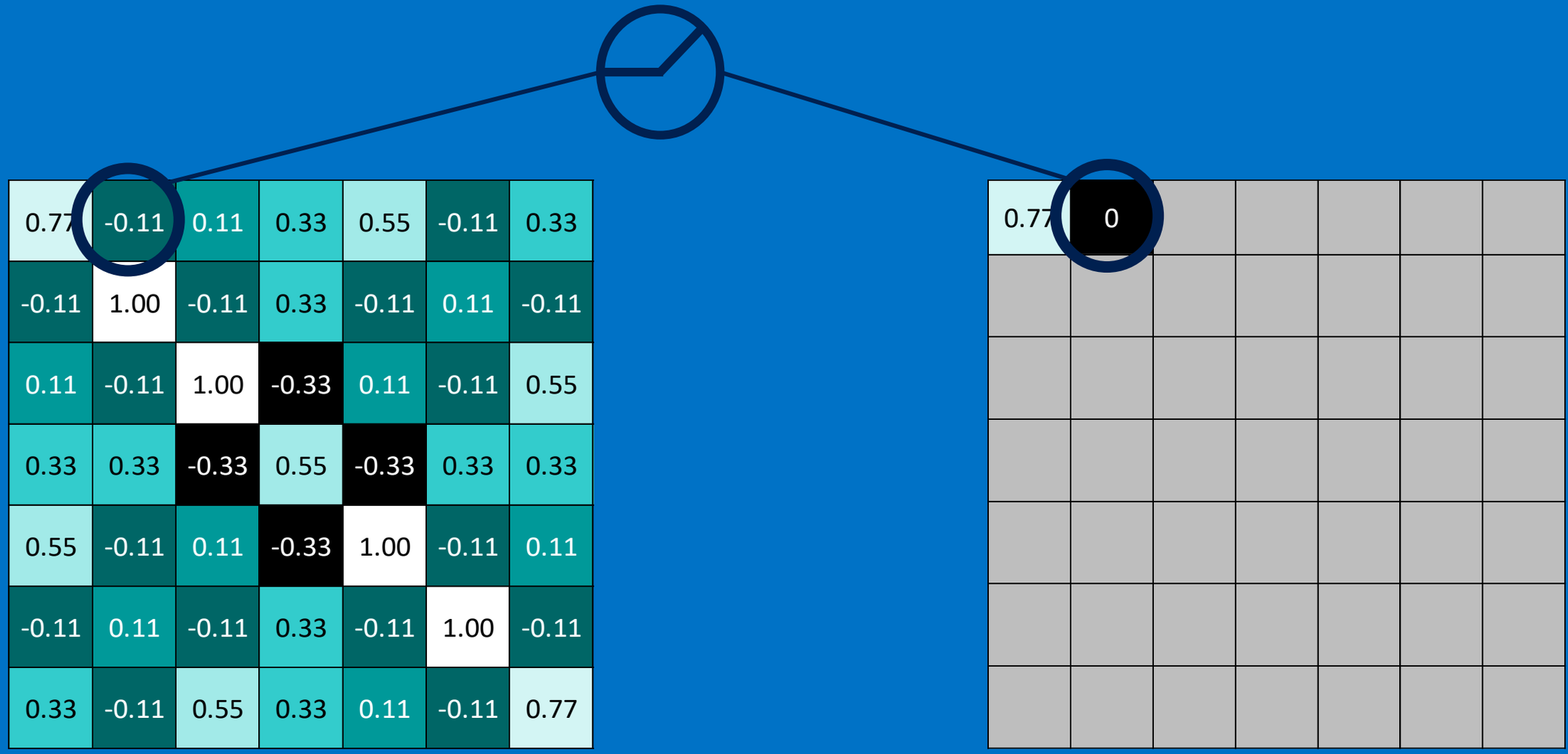


Rectified Linear Units (ReLUs)

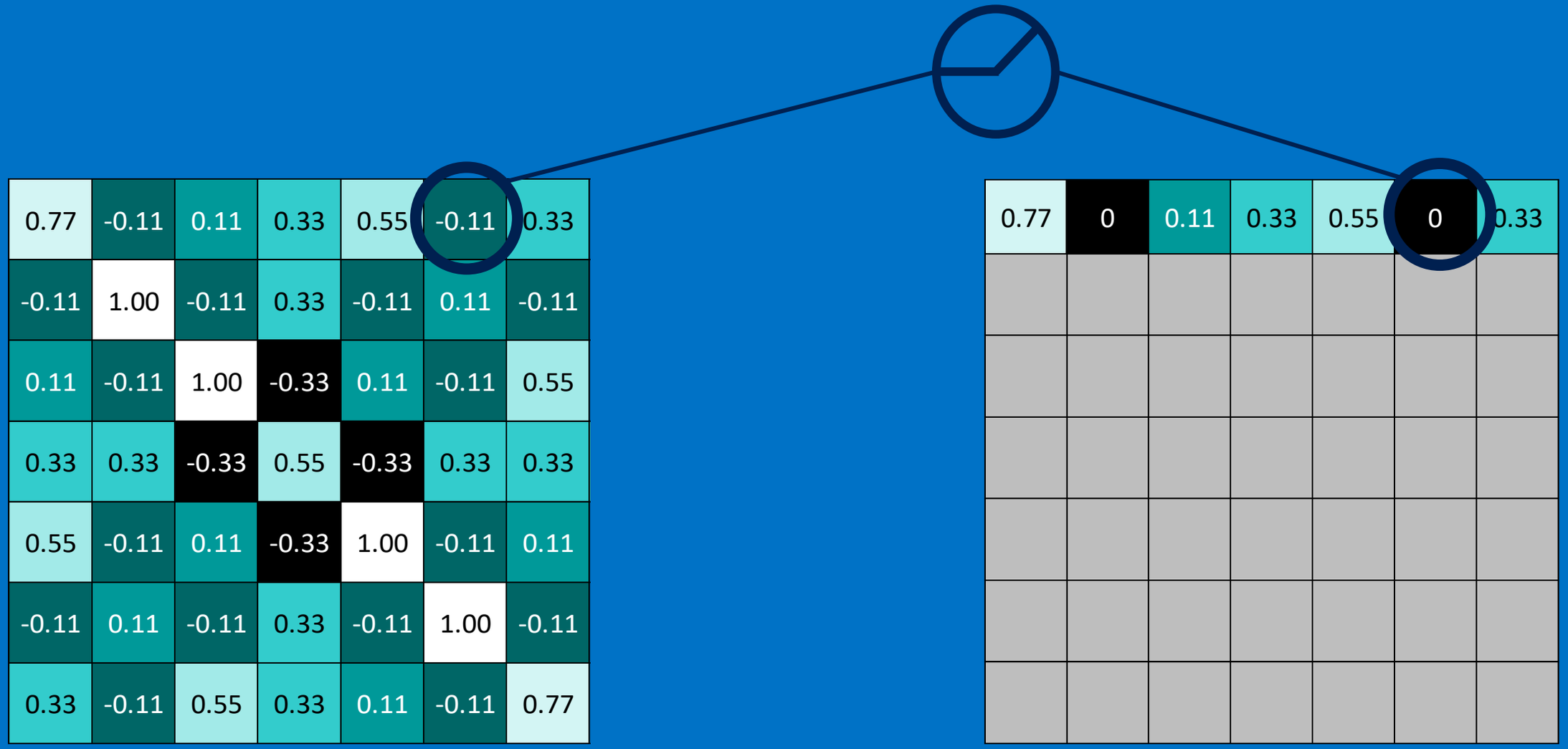


Brandon Rohrer

Rectified Linear Units (ReLUs)



Rectified Linear Units (ReLUs)



Brandon Rohrer

Rectified Linear Units (ReLUs)

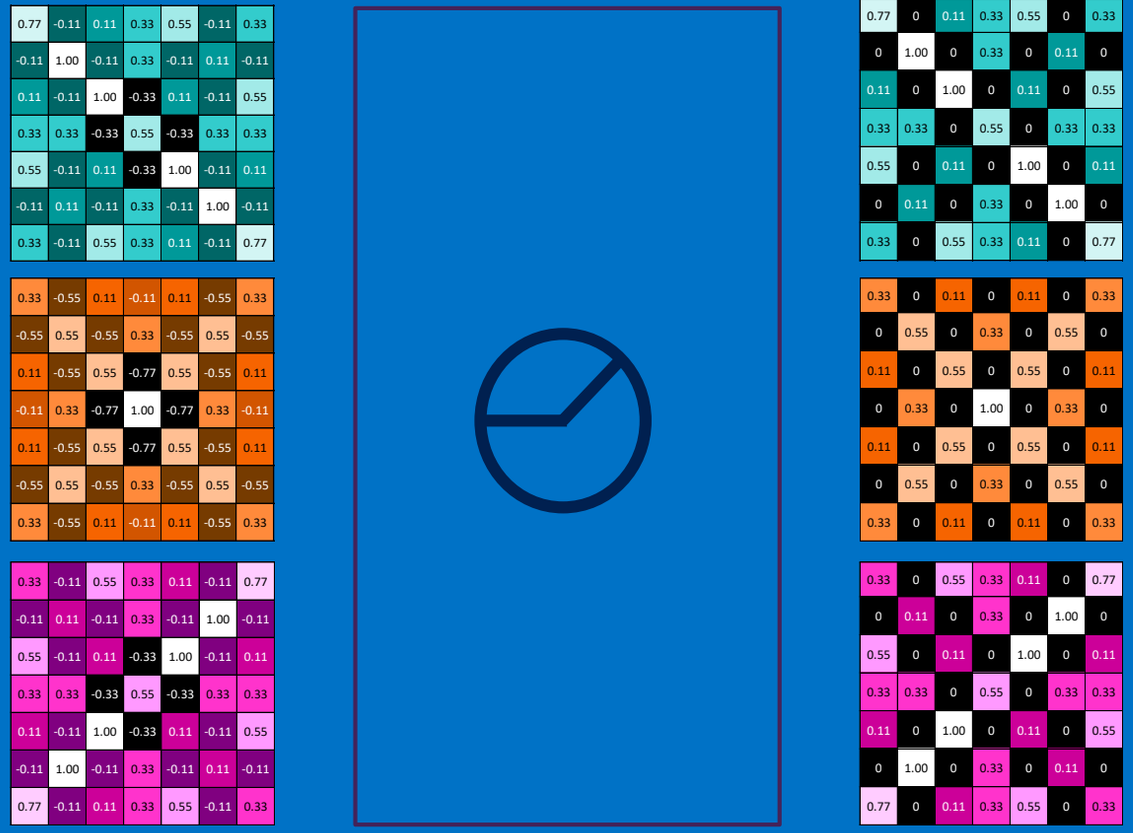
0.77	-0.11	0.11	0.33	0.55	-0.11	0.33
-0.11	1.00	-0.11	0.33	-0.11	0.11	-0.11
0.11	-0.11	1.00	-0.33	0.11	-0.11	0.55
0.33	0.33	-0.33	0.55	-0.33	0.33	0.33
0.55	-0.11	0.11	-0.33	1.00	-0.11	0.11
-0.11	0.11	-0.11	0.33	-0.11	1.00	-0.11
0.33	-0.11	0.55	0.33	0.11	-0.11	0.77

0.77	0	0.11	0.33	0.55	0	0.33
0	1.00	0	0.33	0	0.11	0
0.11	0	1.00	0	0.11	0	0.55
0.33	0.33	0	0.55	0	0.33	0.33
0.55	0	0.11	0	1.00	0	0.11
0	0.11	0	0.33	0	1.00	0
0.33	0	0.55	0.33	0.11	0	0.77

Brandon Rohrer

ReLU layer

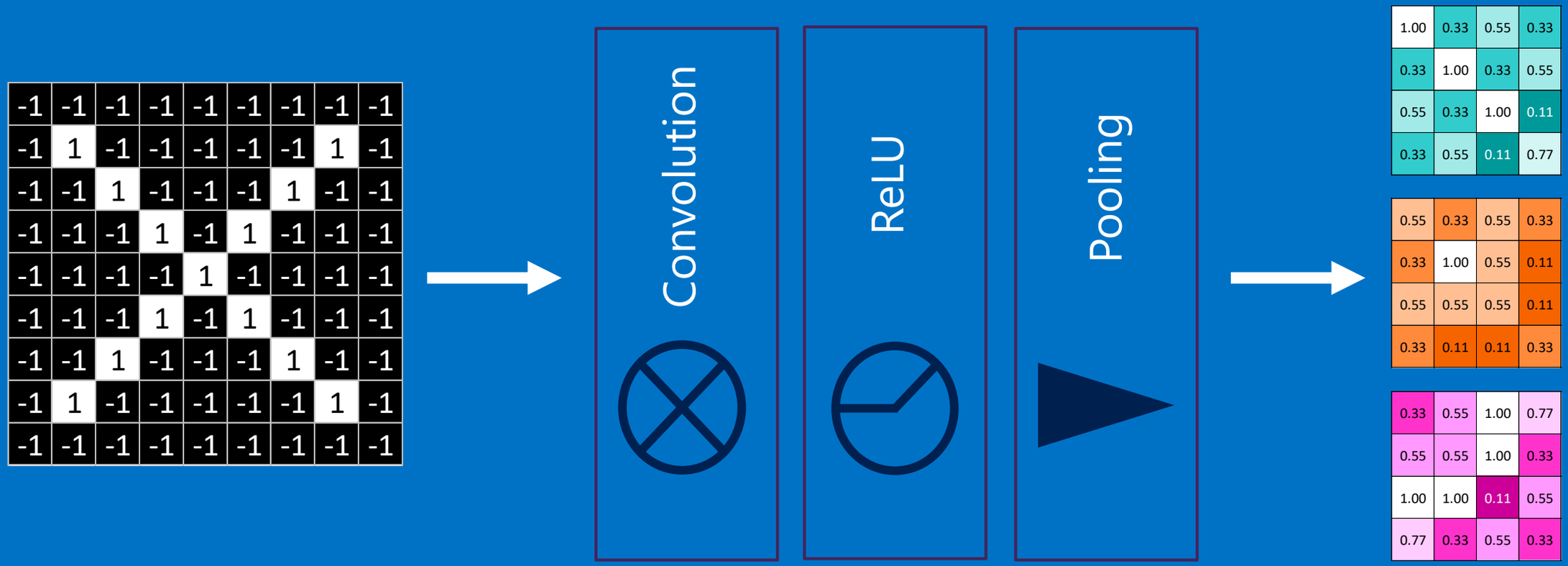
A stack of images becomes a stack of images with no negative values.



Brandon Rohrer

Layers get stacked

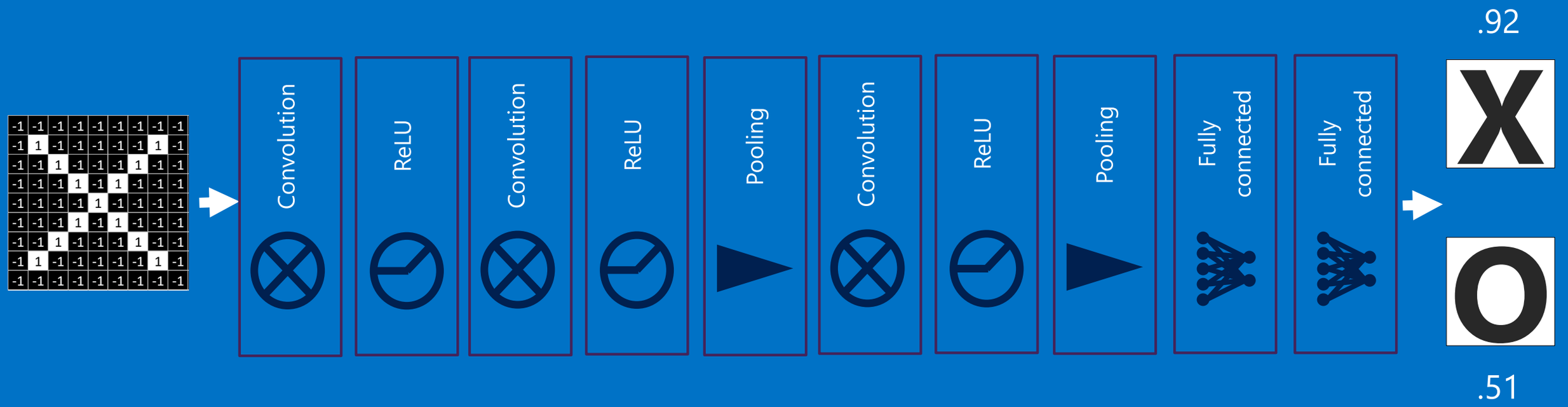
The output of one becomes the input of the next.



Brandon Rohrer

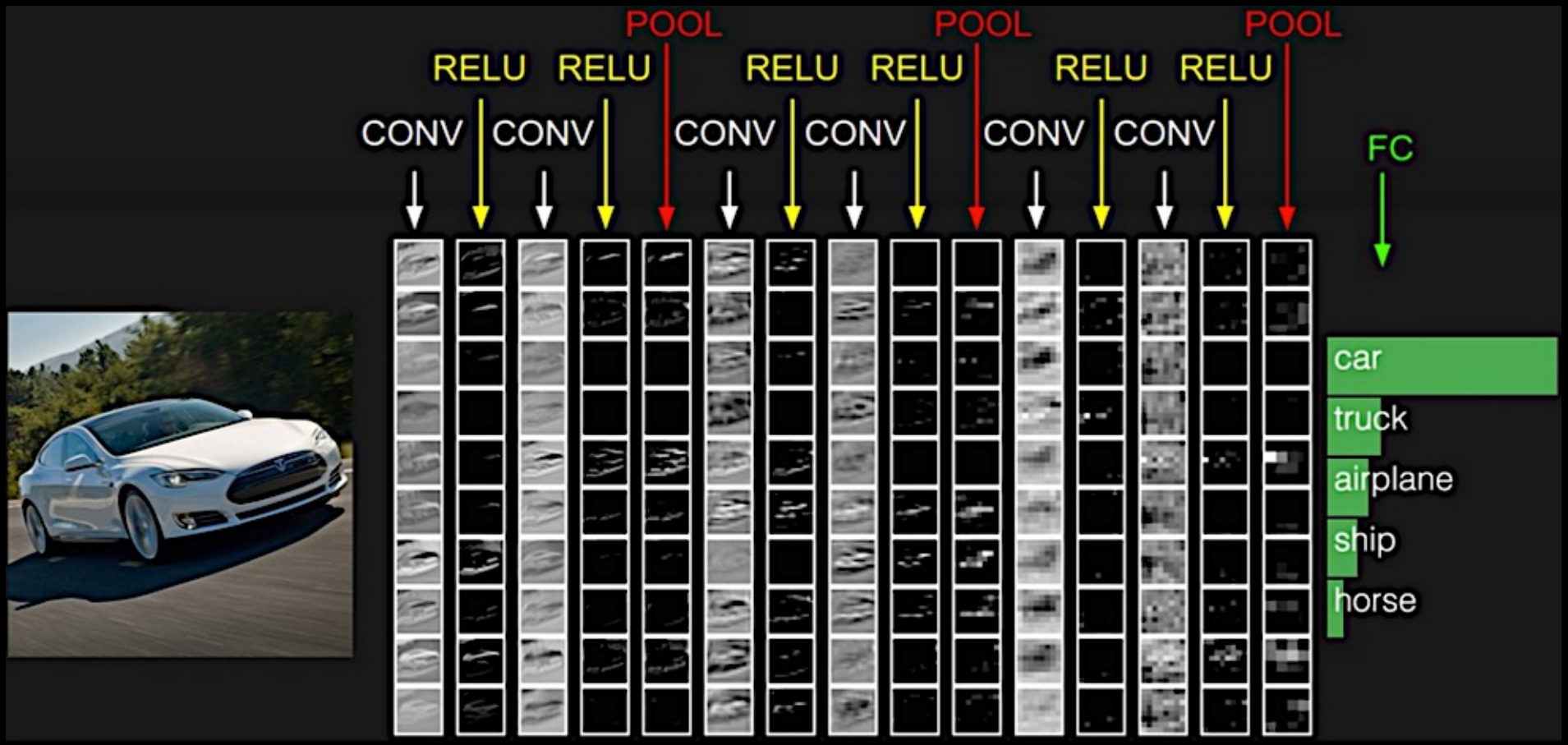
Putting it all together

A set of pixels becomes a set of votes.



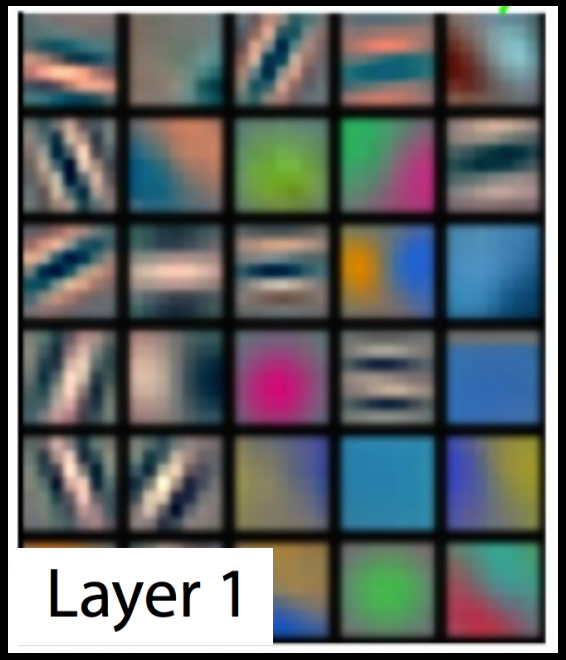
Brandon Rohrer

What do we learn?



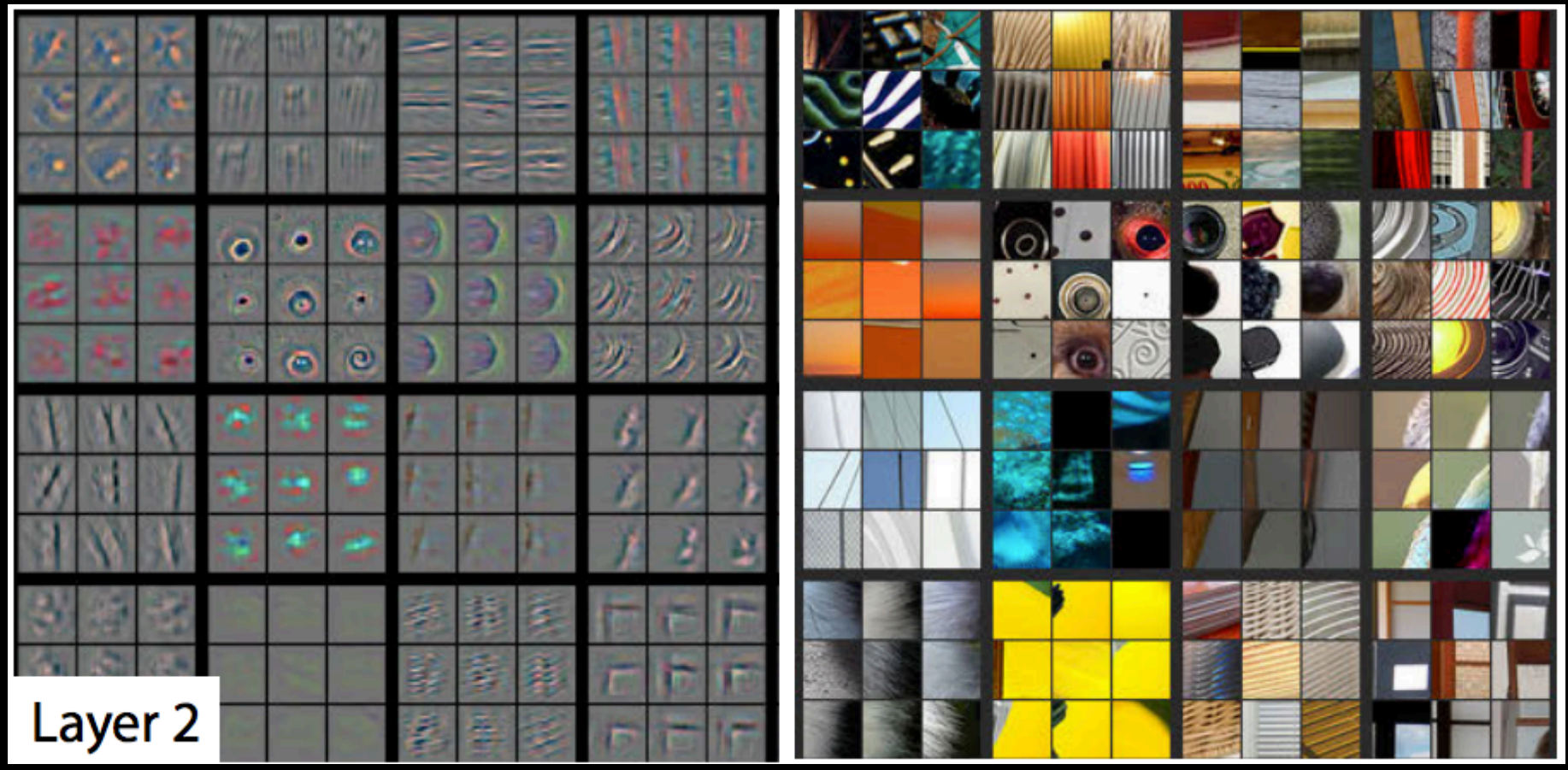
http://cs.stanford.edu/people/karpathy/convnetjs/demo/cifar10.html

What do we learn?



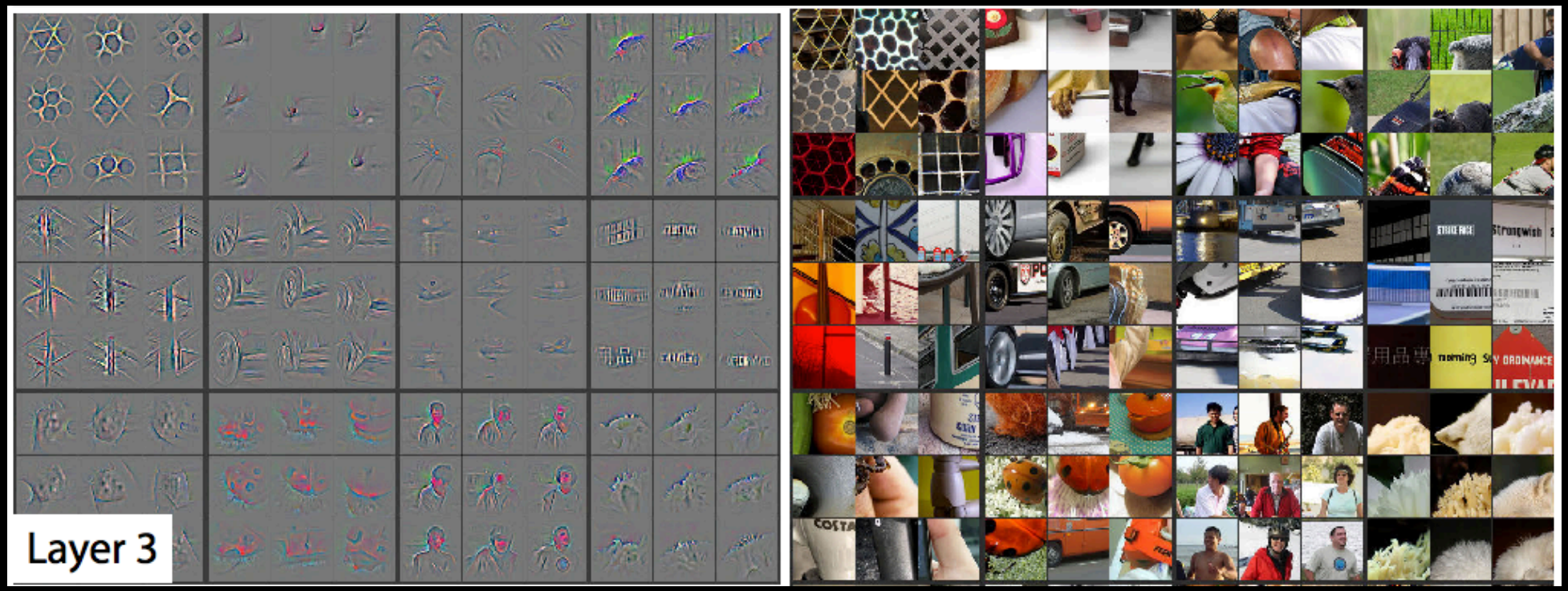
[Zeiler, Fergus, 2013]

What do we learn?



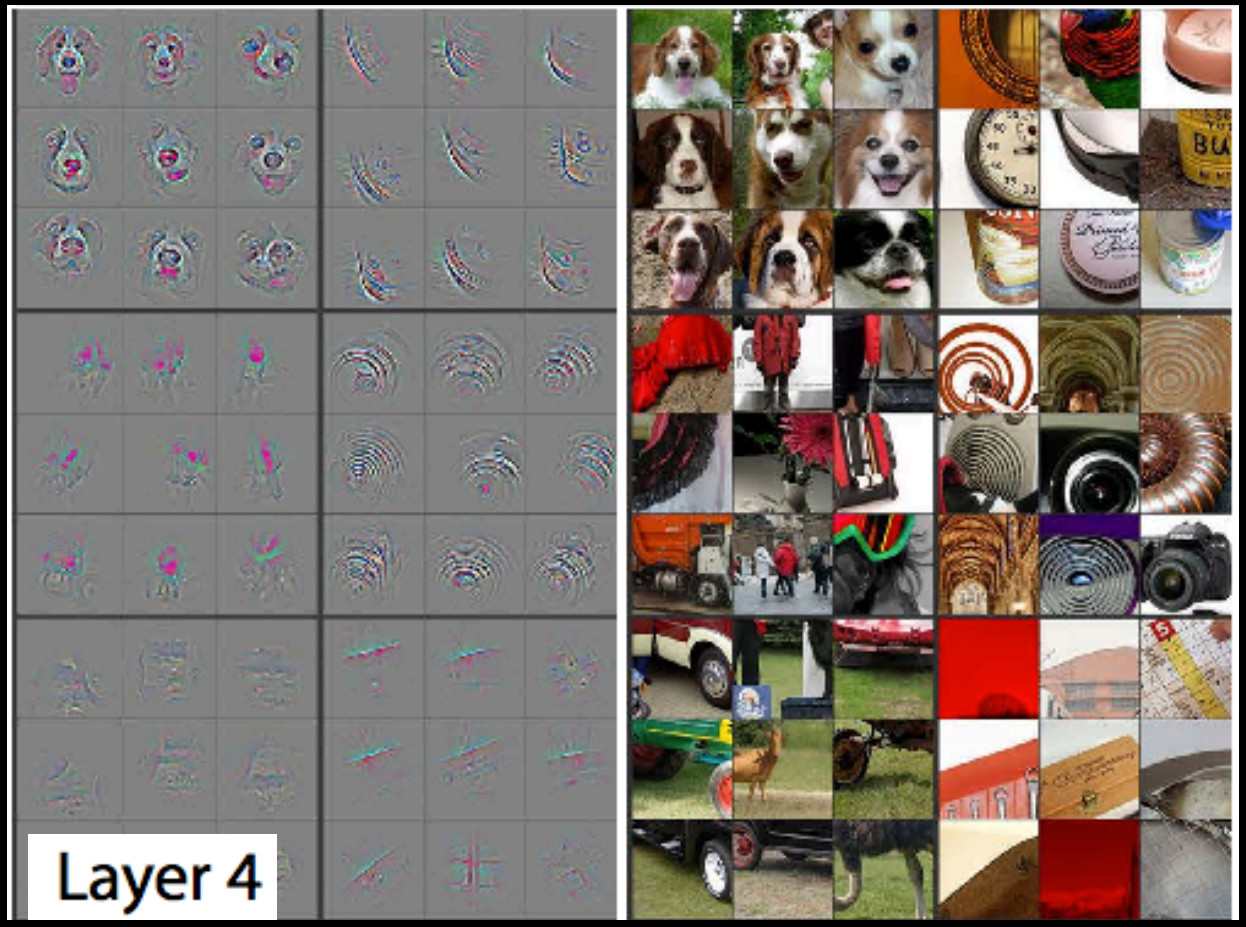
[Zeiler, Fergus, 2013]

What do we learn?



[Zeiler, Fergus, 2013]

What do we learn?



[Zeiler, Fergus, 2013]

Deep Learning - Basics

Usage Requirements

Large data set with good quality *(input-output mappings)*

Measurable and describable goals *(define the cost)*

Enough computing power *(AWS GPU Instance)*

Excels in tasks where the basic unit *(pixel, word)* has very little meaning in itself, but the combination of such units has a useful meaning

Why GPU Matters in Deep Learning?

```
X_train shape: (50000, 3, 32, 32)
50000 train samples
10000 test samples
Using real-time data augmentation.
Epoch 1/200
50000/50000 [==============================] - 734s
Epoch 2/200
50000/50000 [==============================] - 733s
Epoch 3/200
50000/50000 [==============================] - 733s
Epoch 4/200
50000/50000 [==============================] - 733s
```

Running time **without** GPU

vs

```
X_train shape: (50000, 3, 32, 32)
50000 train samples
10000 test samples
Using real-time data augmentation.
Epoch 1/200
50000/50000 [==============================] - 27s
Epoch 2/200
50000/50000 [==============================] - 27s
Epoch 3/200
50000/50000 [==============================] - 27s
Epoch 4/200
50000/50000 [==============================] - 27s
```

Running time **with** GPU

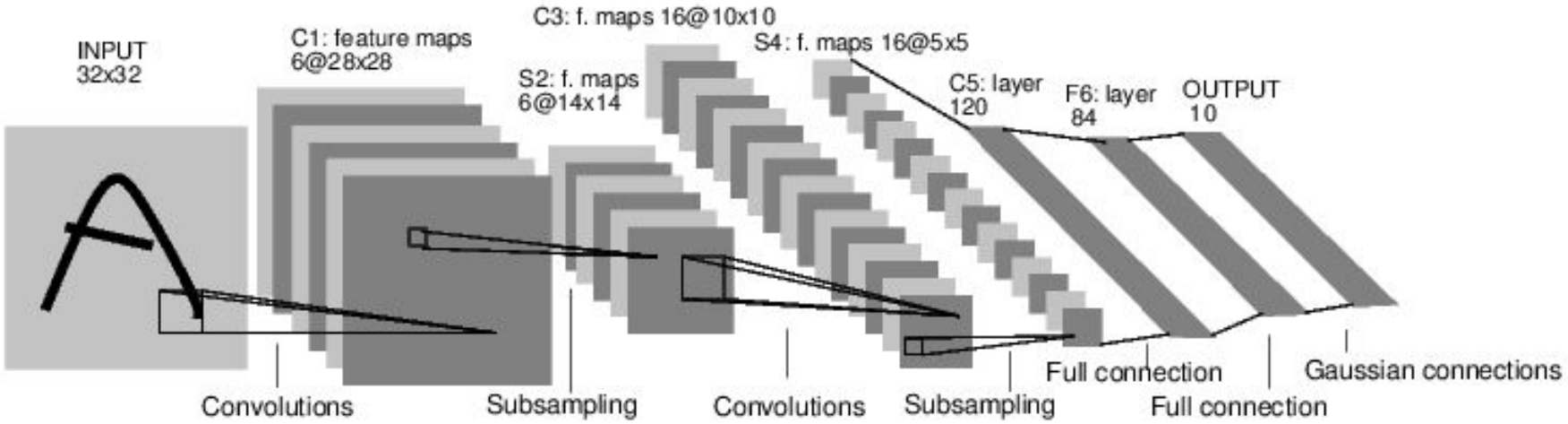
With GPU, the running time is 733/27=**27.1 times faster** then the running time without GPU!!!

Again, WHY GPUs?

1. Every set of weights can be stored as a matrix (m,n)
2. GPUs are made to do common parallel problems fast. All similar calculations are done at the same time. This extremely boosts the performance in parallel computations.

Case Study: LeNet-5

[LeCun et al., 1998]



Conv filters were 5x5, applied at stride 1
Subsampling (Pooling) layers were 2x2 applied at stride 2
i.e. architecture is [CONV-POOL-CONV-POOL-CONV-FC]

ImageNet

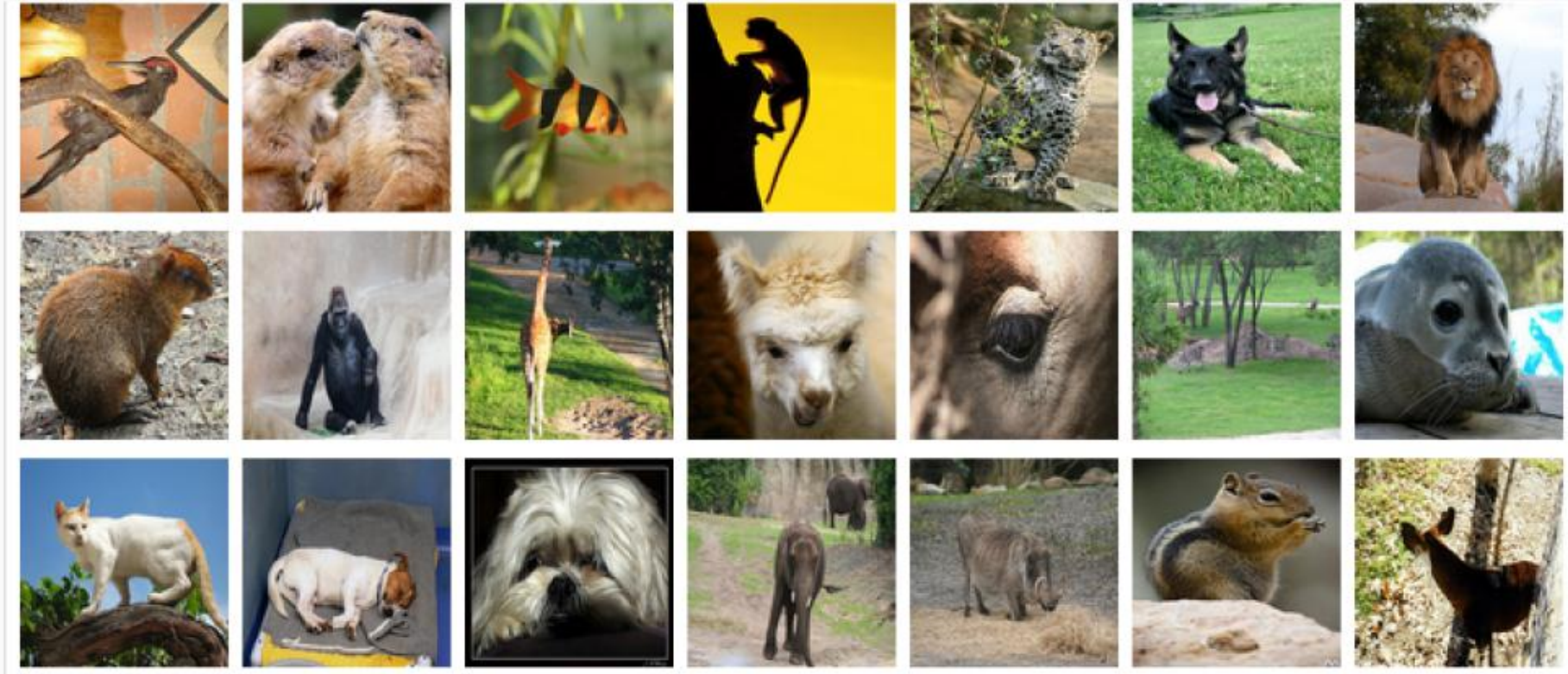
- 15M images
- 22K categories
- Images collected from Web
- Human labelers (Amazon's Mechanical Turk crowd-sourcing)
- ImageNet Large Scale Visual Recognition Challenge (ILSVRC-2010)
  - 1K categories
  - 1.2M training images (~1000 per category)
  - 50,000 validation images
  - 150,000 testing images

- RGB images
- Variable-resolution, but this architecture scales them to 256x256 size

ImageNet

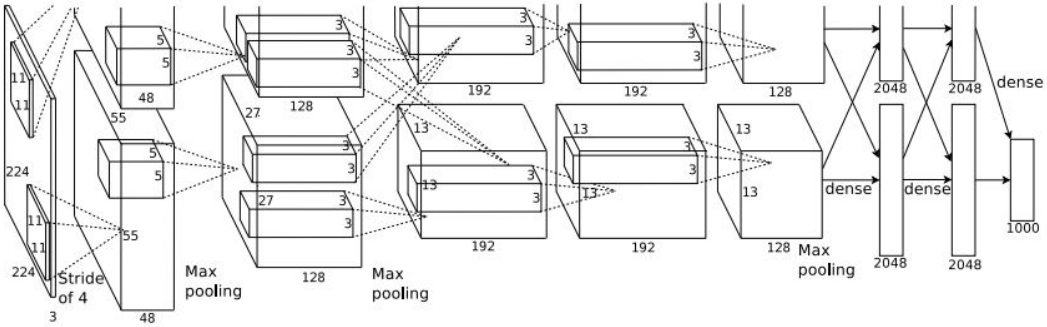
Classification goals:

- ❑ Make 1 guess about the label (Top-1 error)
- ❑ Make 5 guesses about the label (Top-5 error)



Case Study: AlexNet

[Krizhevsky et al. 2012]

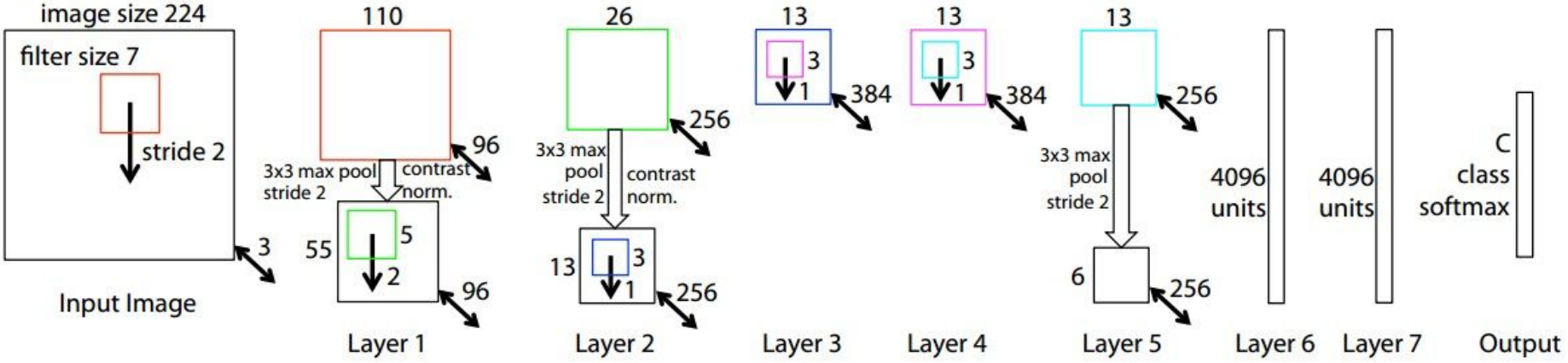


Full (simplified) AlexNet architecture:
[227x227x3] INPUT
[55x55x96] CONV1: 96 11x11 filters at stride 4, pad 0
[27x27x96] MAX POOL1: 3x3 filters at stride 2
[27x27x96] NORM1: Normalization layer
[27x27x256] CONV2: 256 5x5 filters at stride 1, pad 2
[13x13x256] MAX POOL2: 3x3 filters at stride 2
[13x13x256] 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%

Case Study: ZFNet

[Zeiler and Fergus, 2013]



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

ImageNet top 5 error: 15.4% -> 14.8%

Case Study: VGGNet

[Simonyan and Zisserman, 2014]

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

best model

11.2% top 5 error in ILSVRC 2013
->
7.3% top 5 error

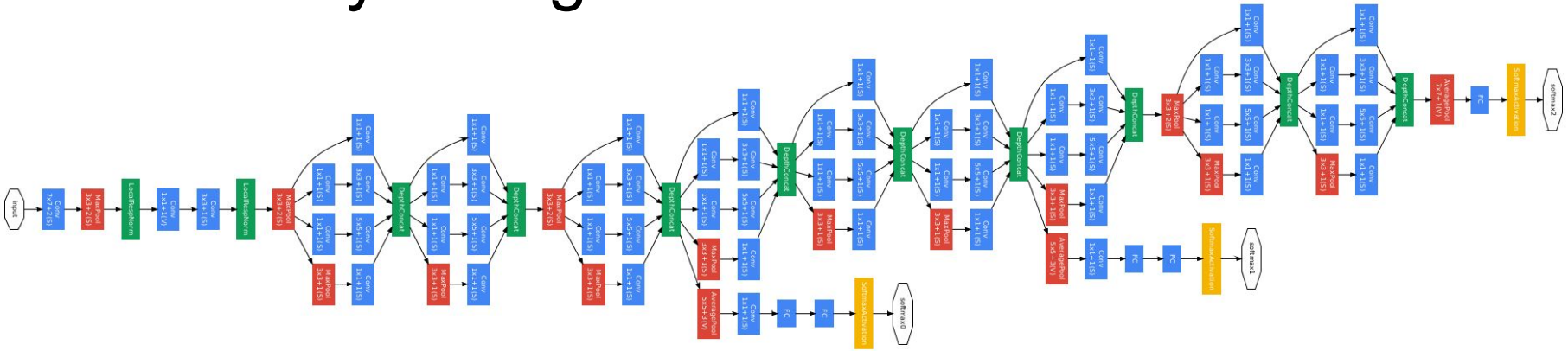
ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	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 **conv1-256**	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 **conv1-512**	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 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

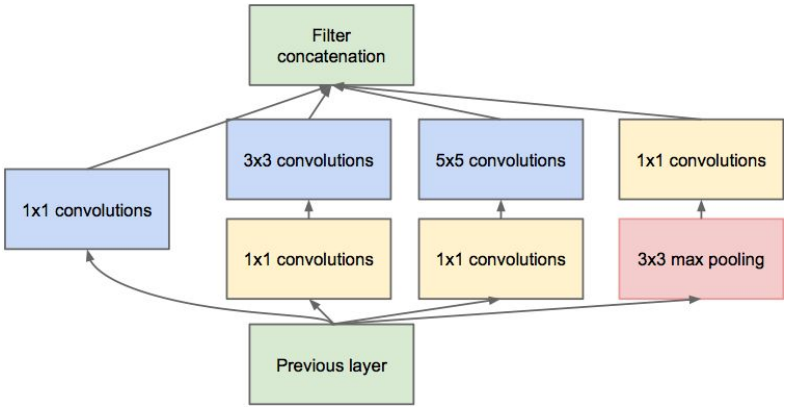
Table 2: **Number of parameters** (in millions).

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

Case Study: GoogLeNet

[Szegedy et al., 2014]



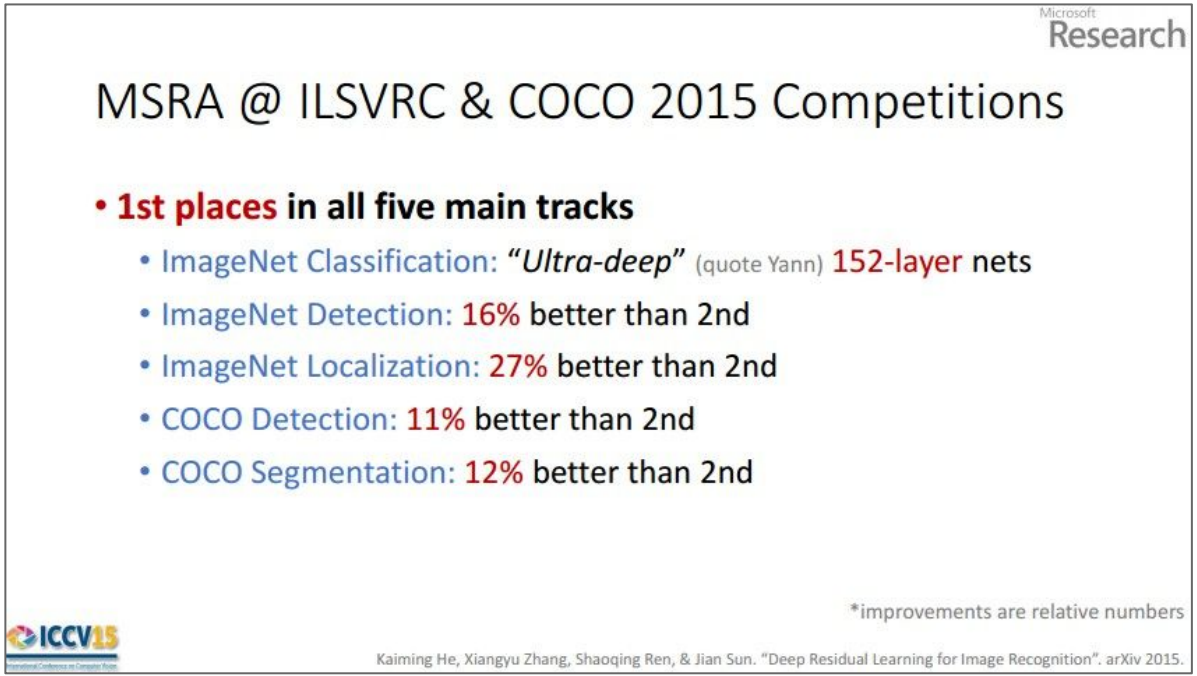


Inception module

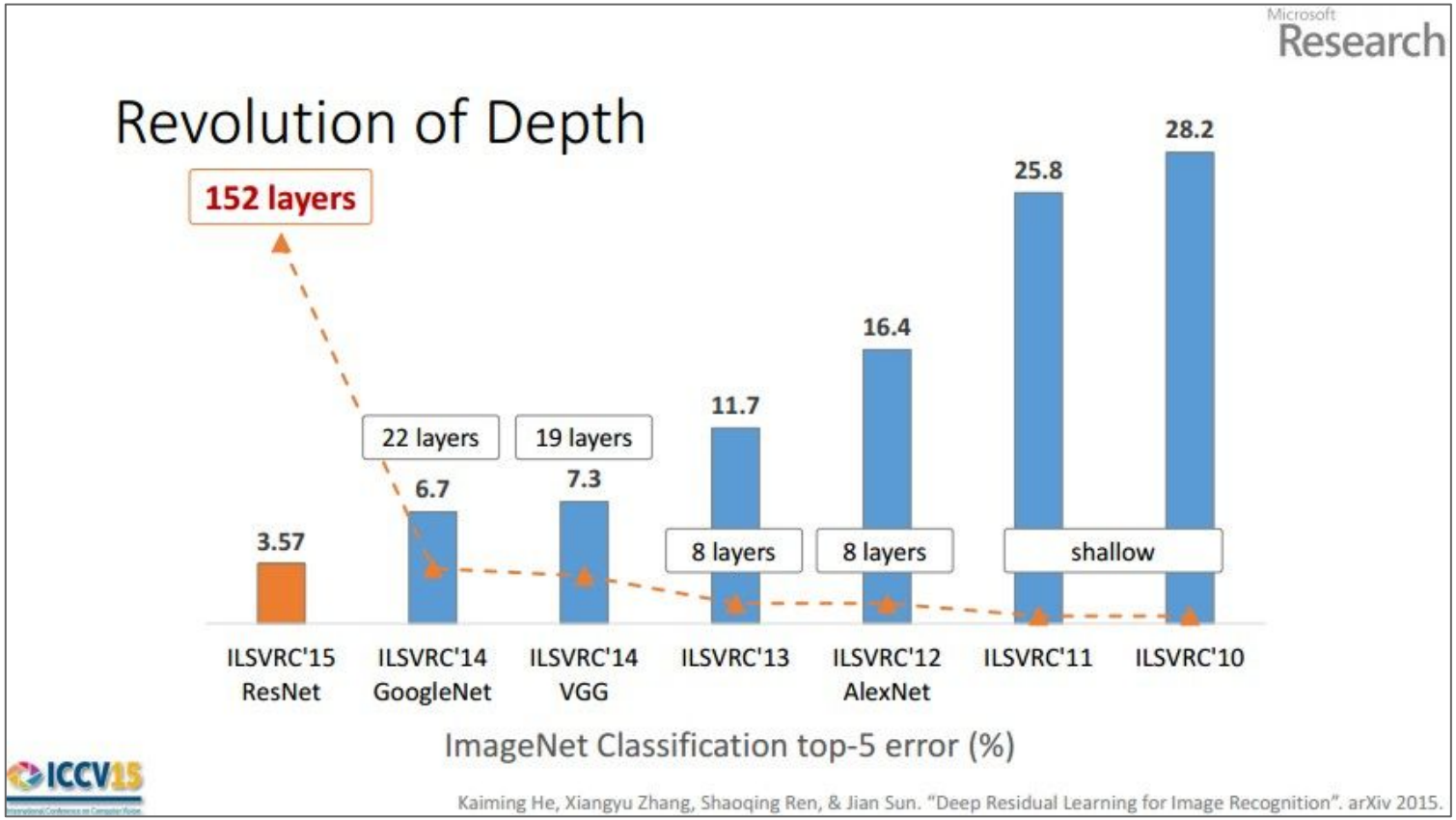
ILSVRC 2014 winner (6.7% top 5 error)

Case Study: ResNet *[He et al., 2015]*

ILSVRC 2015 winner (3.6% top 5 error)



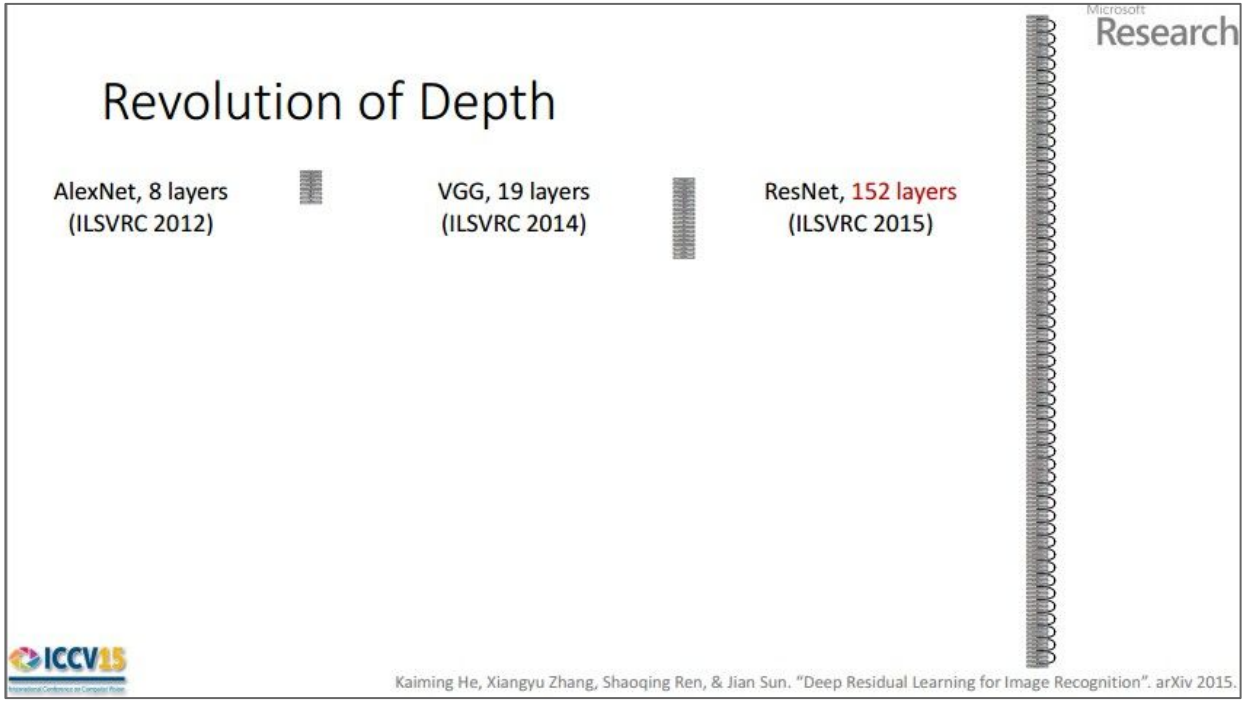
Slide from Kaiming He's recent presentation https://www.youtube.com/watch?v=1PGLj-uKT1w



(slide from Kaiming He's recent presentation)

Case Study: ResNet *[He et al., 2015]*

ILSVRC 2015 winner (3.6% top 5 error)



(slide from Kaiming He's recent presentation)

2-3 weeks of training on 8 GPU machine

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

Audio

TIMIT Phone classification	Accuracy
Prior art (Clarkson et al.,1999)	79.6%
Stanford Feature learning	**80.3%**

TIMIT Speaker identification	Accuracy
Prior art (Reynolds, 1995)	99.7%
Stanford Feature learning	**100.0%**

Images

CIFAR Object classification	Accuracy
Prior art (Yu and Zhang, 2010)	74.5%
Stanford Feature learning	**75.5%**

NORB Object classification	Accuracy
Prior art (Ranzato et al., 2009)	94.4%
Stanford Feature learning	**96.2%**

Video

UCF activity classification	Accuracy
Prior art (Kalser et al., 2008)	86%
Stanford Feature learning	**87%**

Hollywood2 classification	Accuracy
Prior art (Laptev, 2004)	47%
Stanford Feature learning	**50%**

Multimodal (audio/video)

AVLetters Lip reading	Accuracy
Prior art (Zhao et al., 2009)	58.9%
Stanford Feature learning	**63.1%**