Session - 1

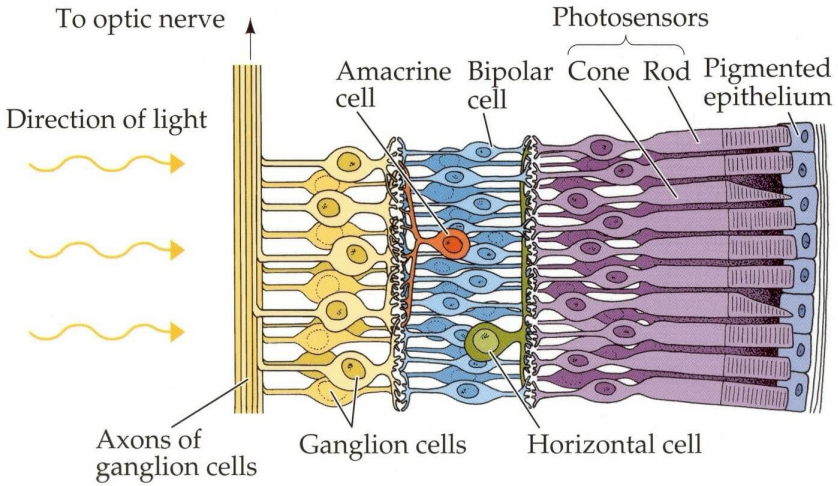
Intro To CNN

March 11, 2024

Agenda

1. Bit of human vision
2. Channels
3. How CV started.
4. Convolution - Filters
5. Max Pooling

How Our Eyes Work



R → 0-255

G → 0-255

B → 0-255

$255 \times 255 \times 255$

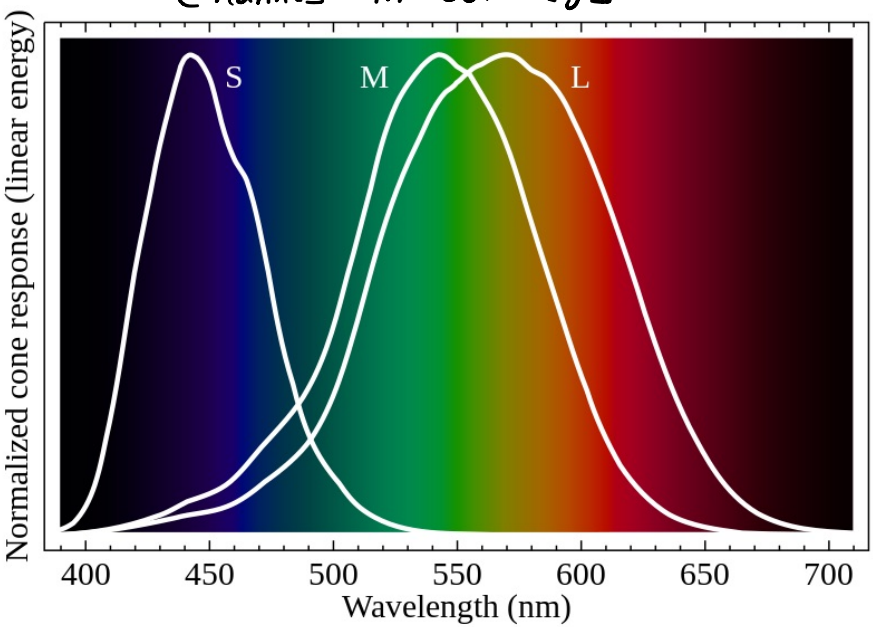
$256 = 2^8$

$2^8 \times 2^8 \times 2^8$

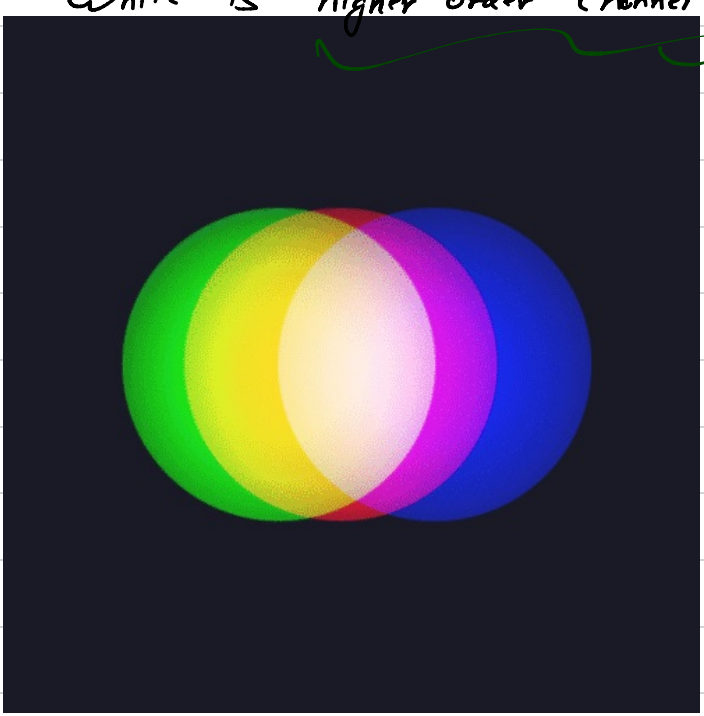
$= 2^{24}$

Channel → Container for similar information

Channels in our Eyes

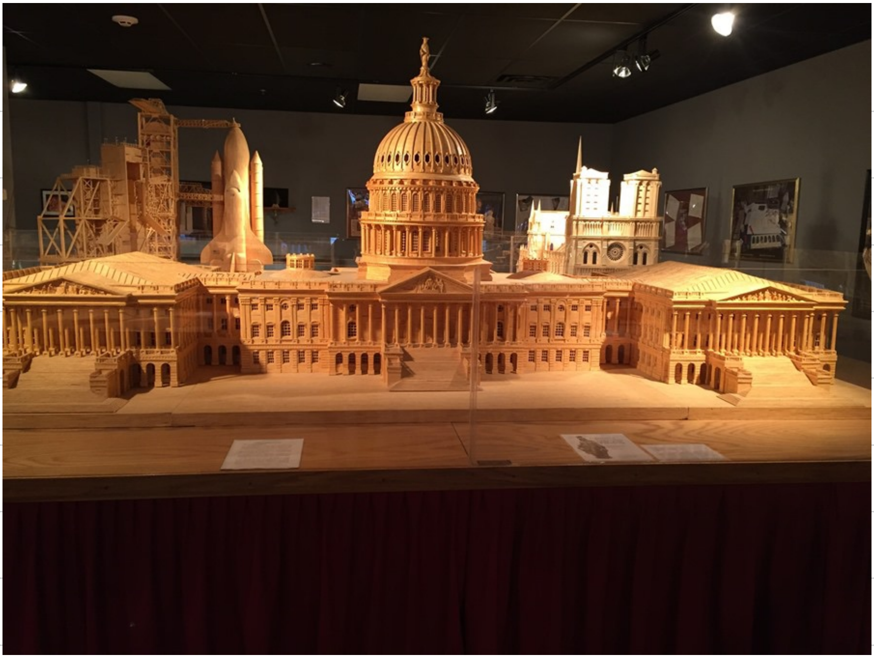


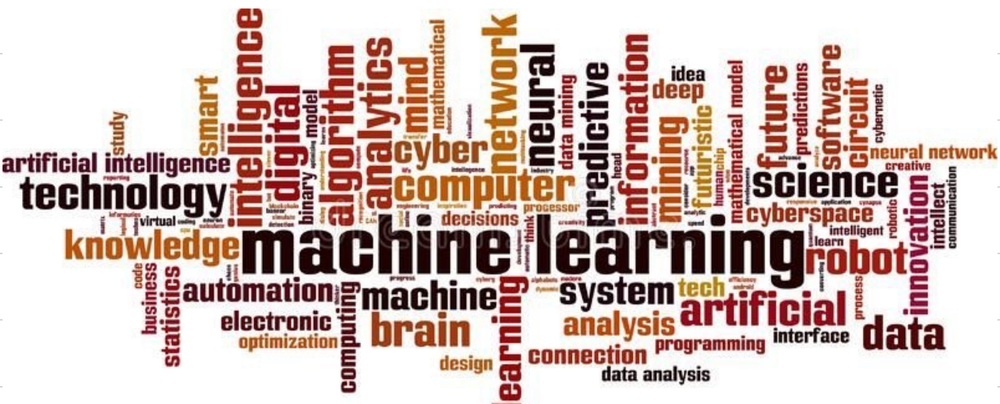
White is higher order channel



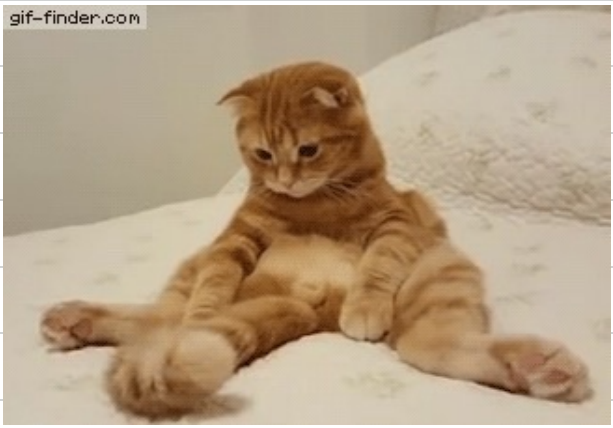
→ Point of reference as RGB

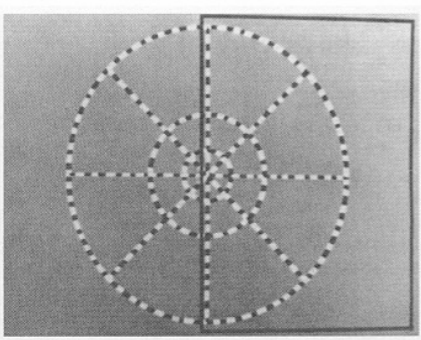
Channels here:



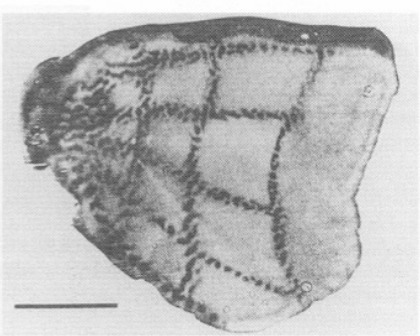

machine learning
artificial intelligence
technology
knowledge
intelligence
digital
algorithm
analytics
mind
mathematical
network
neural
data mining
predictive
information
idea
deep
mining
futuristic
mathematical model
future
predictions
software
circuit
cybernetic
neural network
creative
science
cyberspace
intelligent
robotic
intellect
communication
innovation
robot
learn
cyber
computer
processor
decisions
smart
study
binary
model
virtual
automation
machine
system
tech
artificial
data
interface
programming
analysis
connection
data analysis
design
learning
brain
computing
electronic
optimization
statistics
business
code
humanchip
head
think
process

Gruesome Cat Exp

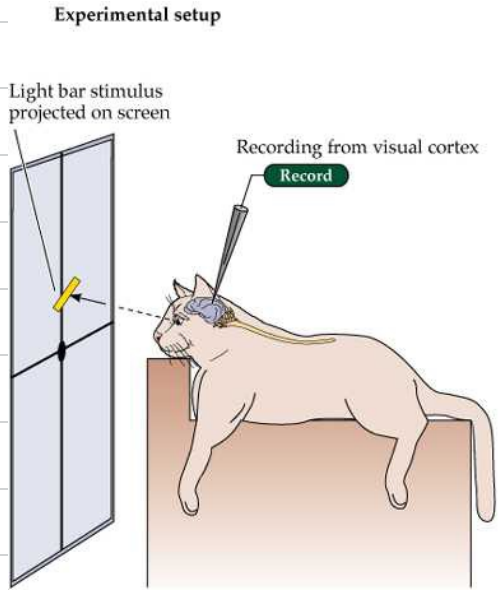


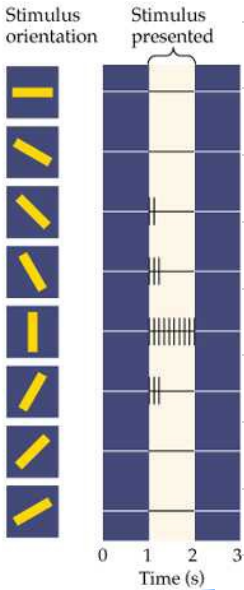


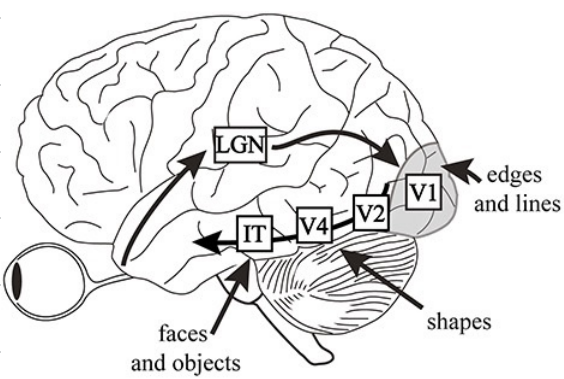
(a)

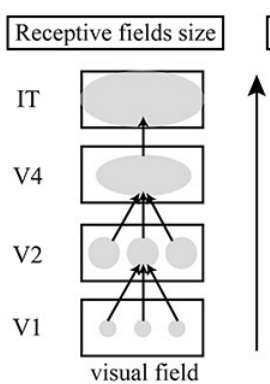


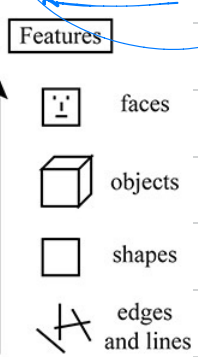
(b)


Experimental setup
Light bar stimulus projected on screen
Recording from visual cortex
Record

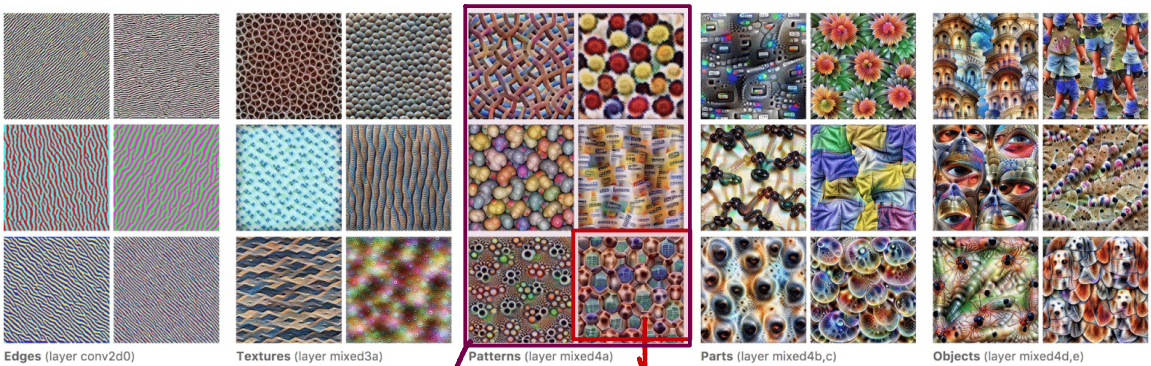

Stimulus orientation
Stimulus presented
0
1
2
3
Time (s)

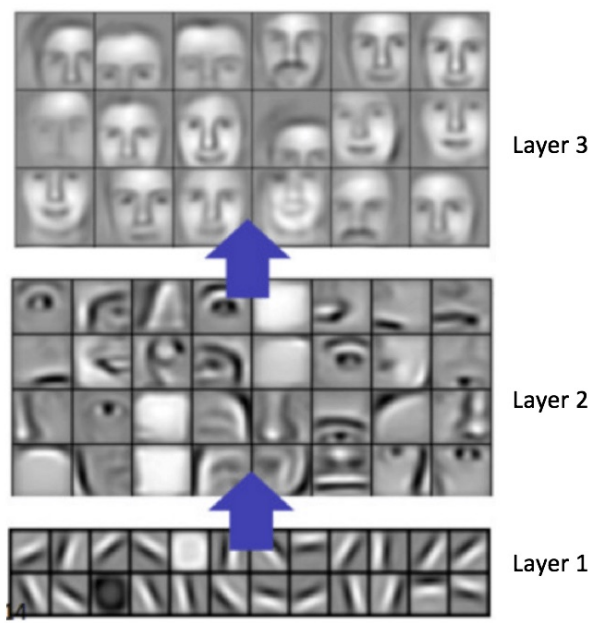

LGN
V1
V2
V4
IT
edges and lines
shapes
faces and objects


Receptive fields size
IT
V4
V2
V1
visual field


Features
faces
objects
shapes
edges and lines

How channels look in each layer

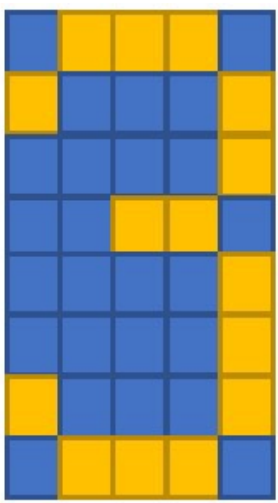




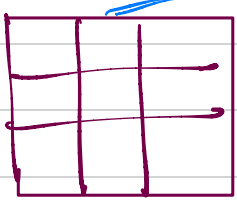
How to maintain spatial information in Neural Network.

What no. is this?? ↑↑ Pixels >>

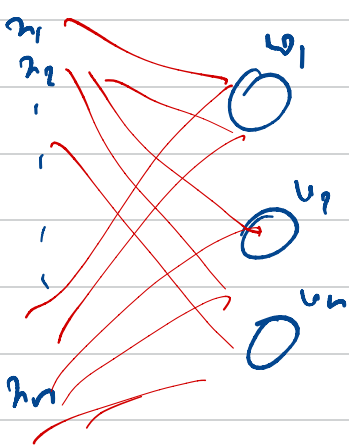
Preserve structure



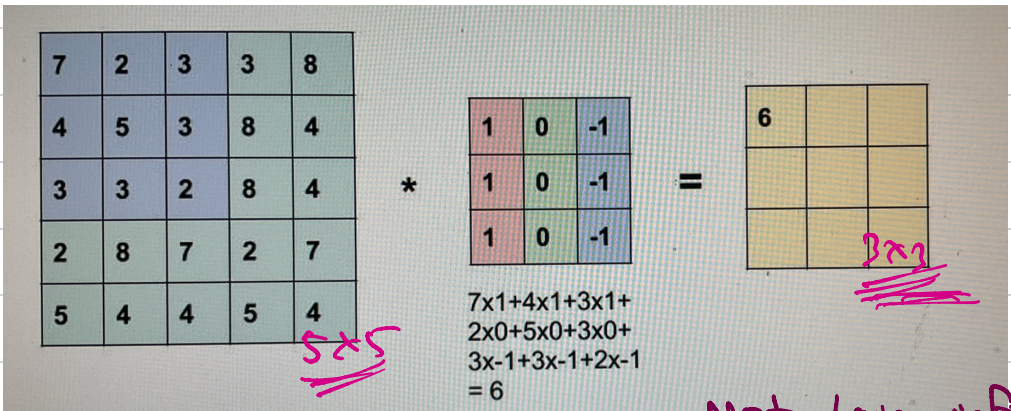
3×3



Filter
or
wt matrix
or
Feature extractor.
or
Kernel

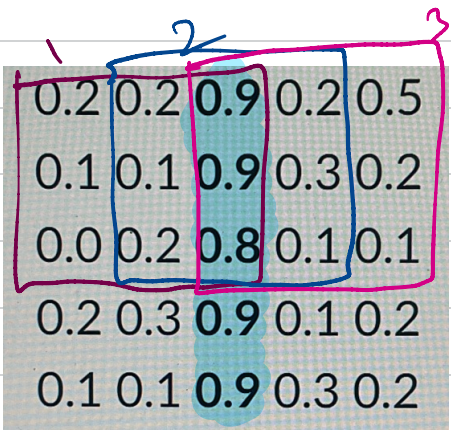


Solution: Convolution



Not losing info / filtering

How Convolution works??



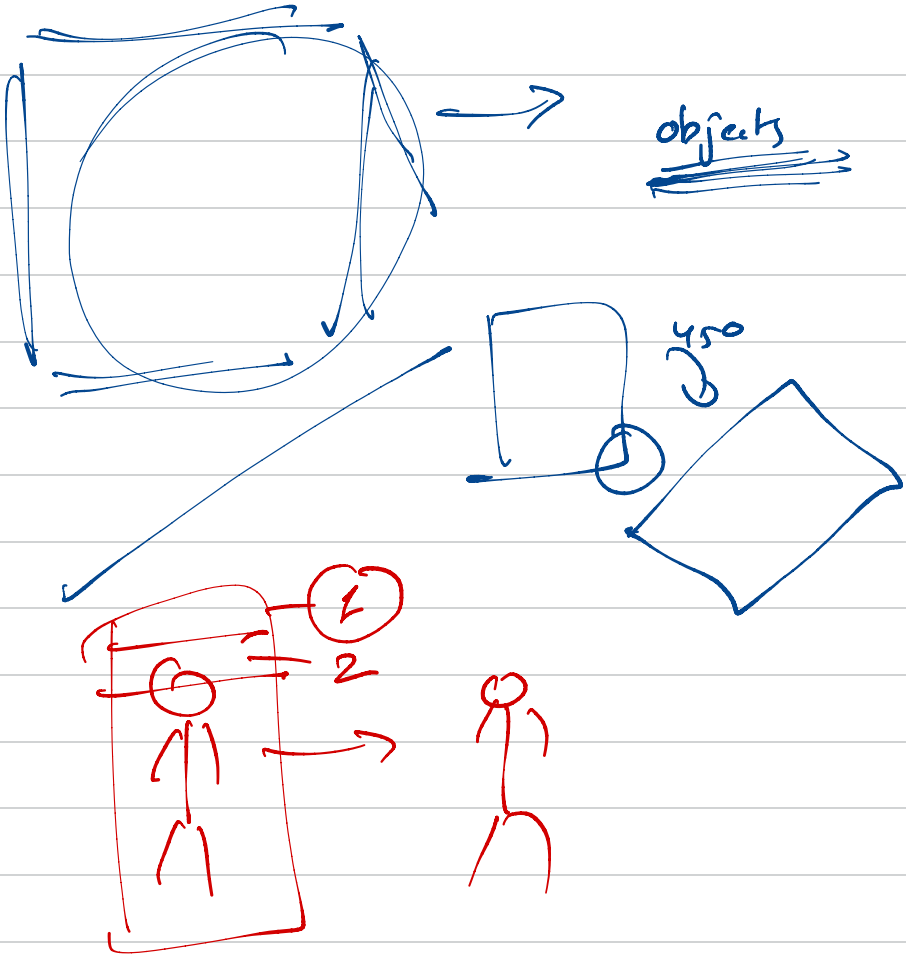
$$\cdot \begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix}$$

Feature extractor for Vertical Edges

- daric dar dast

$$\Downarrow$$

$$\begin{bmatrix} -1.9 & 4.3 & -2.3 \\ -1.7 & 4.1 & -2.1 \\ -1.7 & 4.1 & -2.1 \end{bmatrix}$$



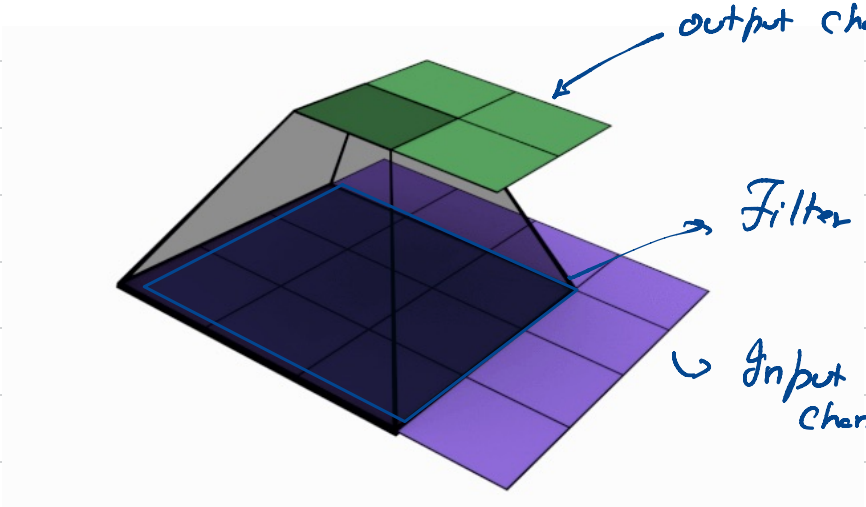
Yann - Lecun → 1989 LeNet

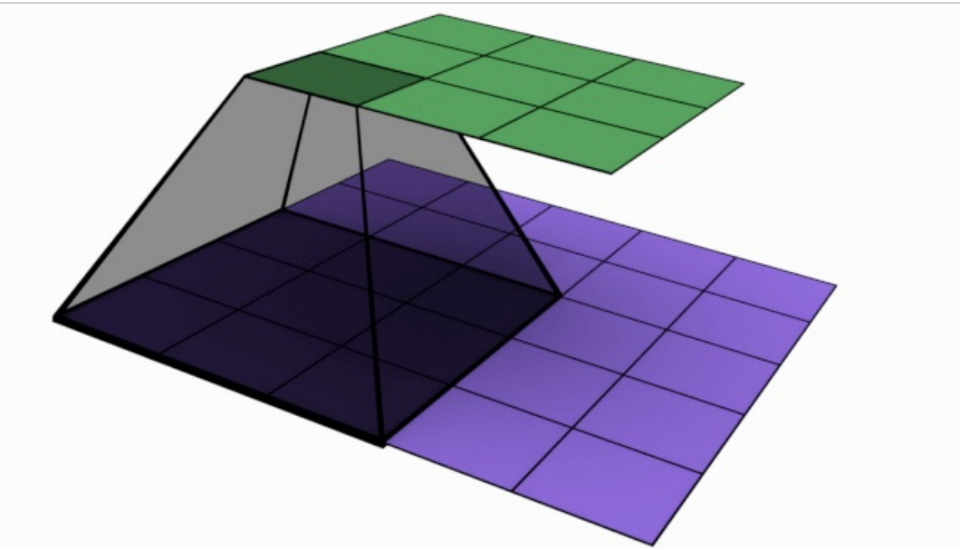
2012

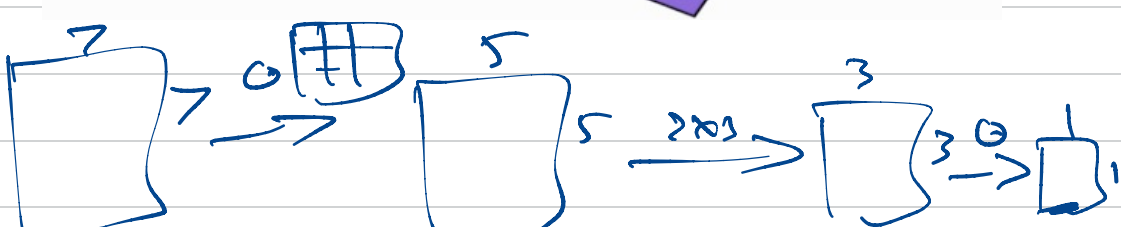
Alex Krc

CV

2 → Alex Net

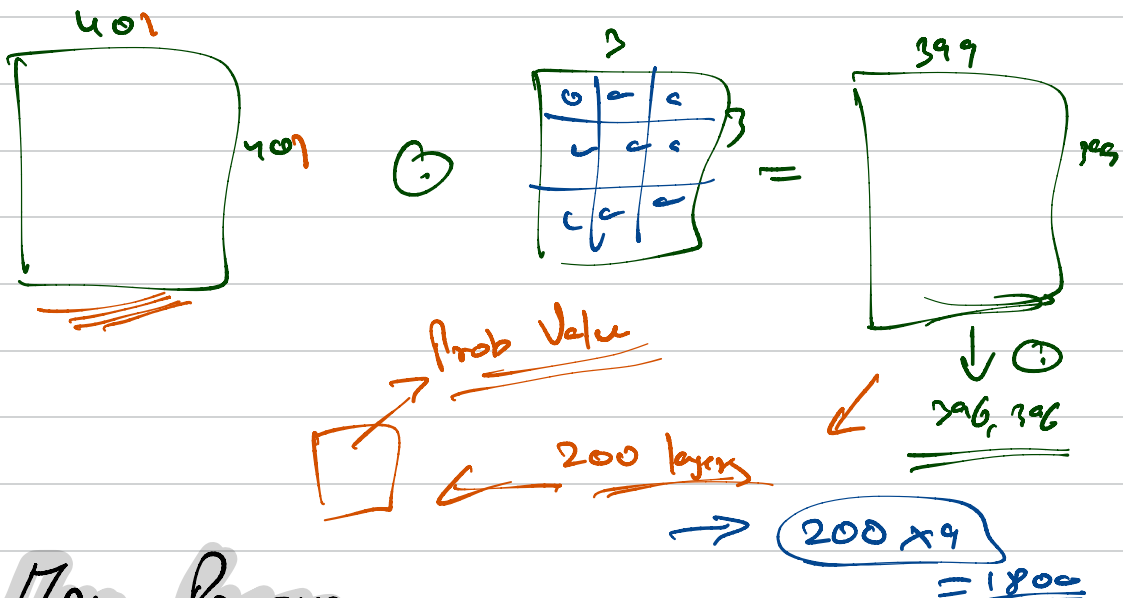

output Channel
Filter
Input
Channel



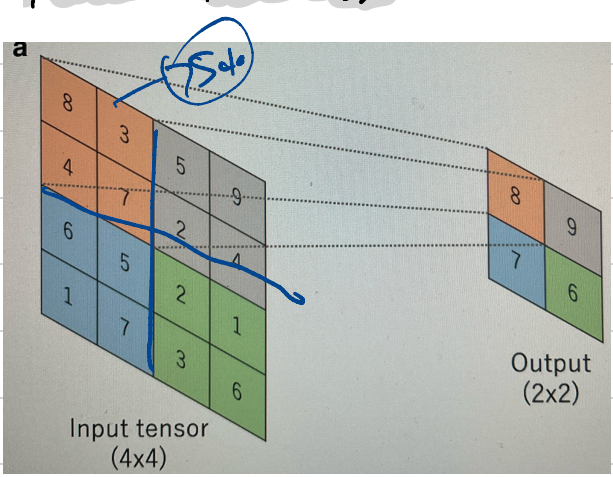

7
5
5
3
3
1
1
2x2

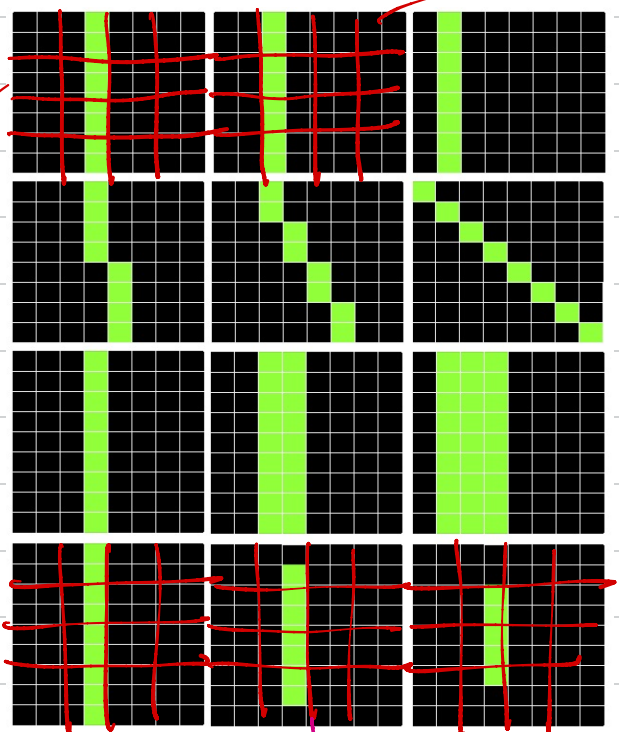
→ Kernels act as Feature Extractors !!

→ How many layers would we need to move from 401 x 401 image to 1x1 !!



MAX POOLING

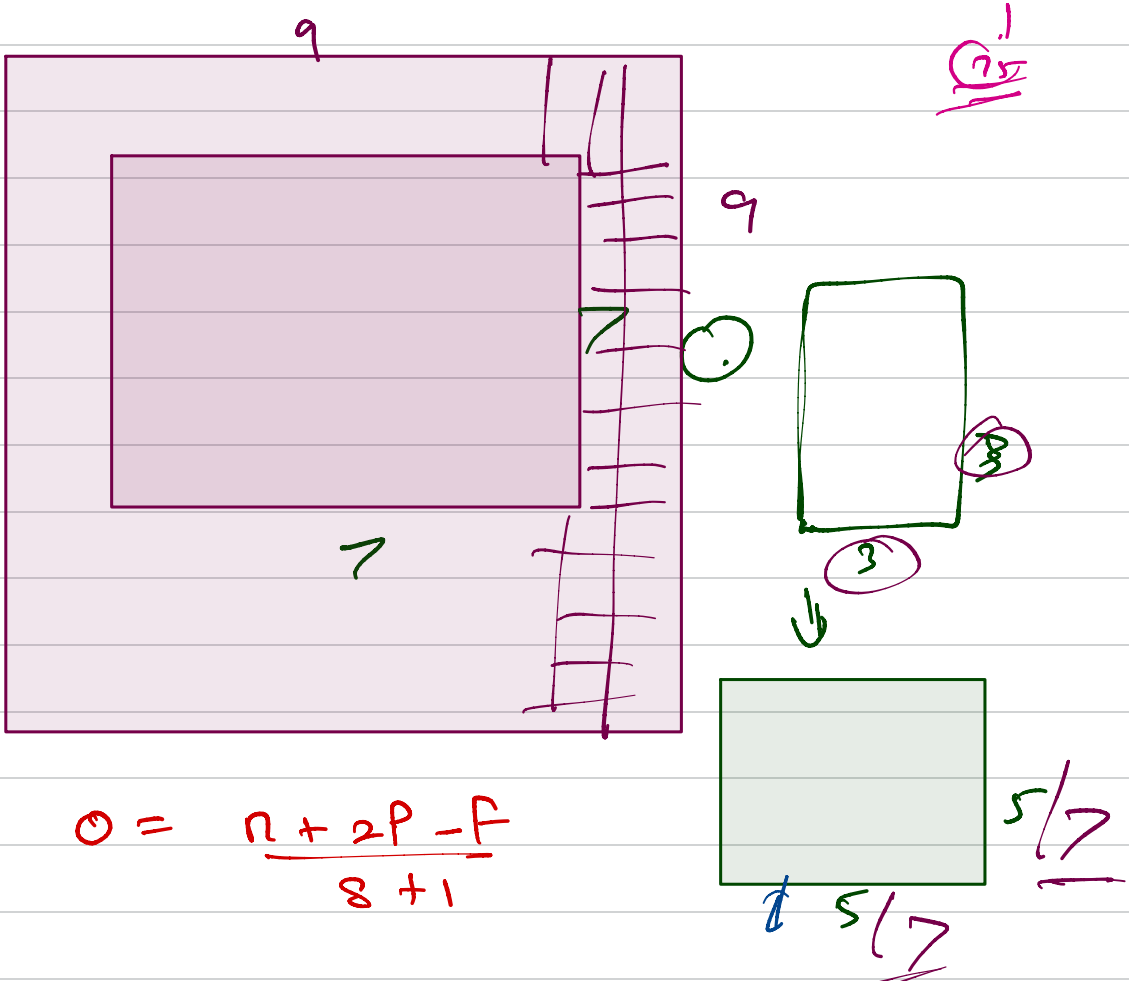




mean Boolean adjusts for shifting also adjusts for rotation

400	398	396	394	392	390	MP (2x2)
195	193	191	189	187	185	MP (2x2)
92	90	88	86	84	82	MP (2x2)
41	39	37	35	33	31	MP (2x2)
15	13	11	9	7	5	3	1	→ 27

400 → 200 → 100 → 50



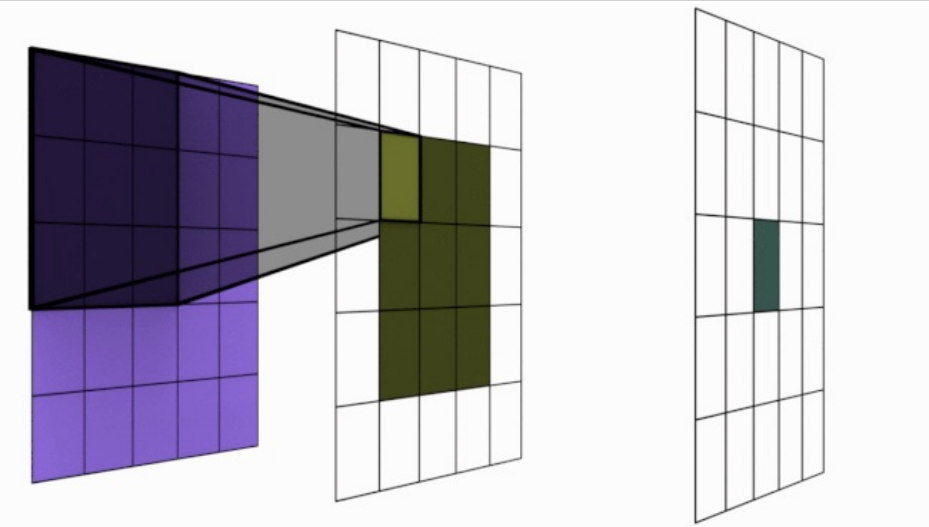
$$O = \frac{n + 2P - F}{S + 1}$$

In tf/keras

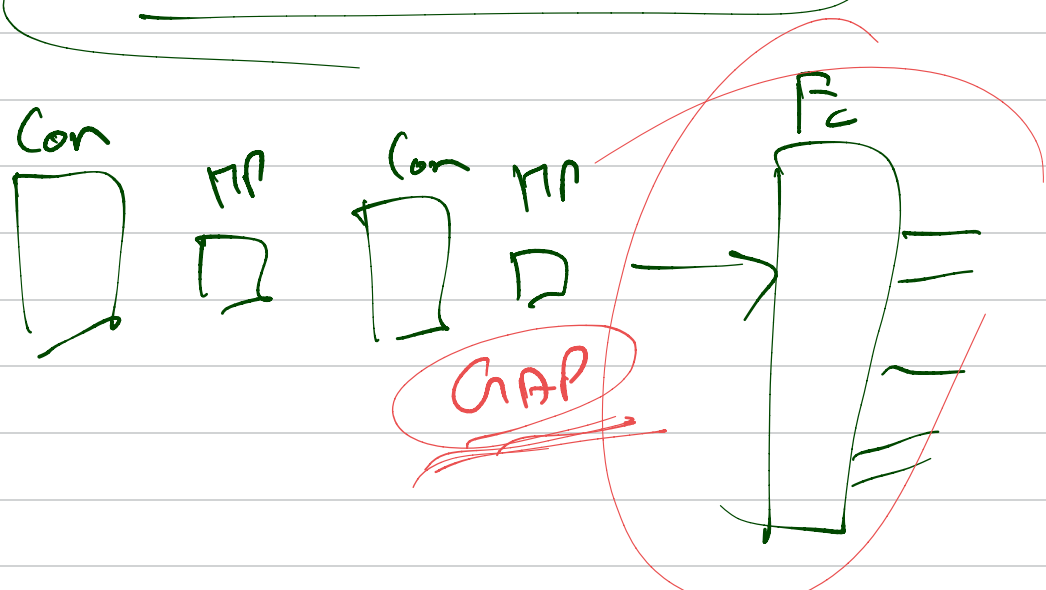
default Padding → 0/1 (Not recommended)

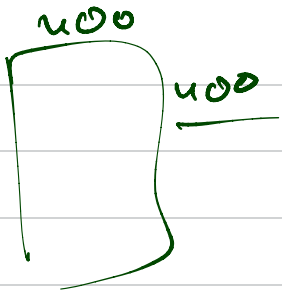
↳ Pad with avg pixel value

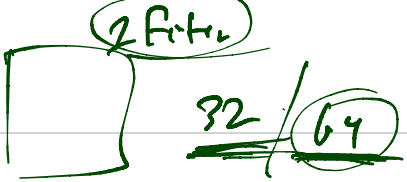
↳ " " last pixel value of edge

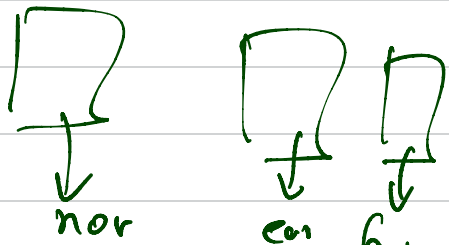


Global Average Pooling

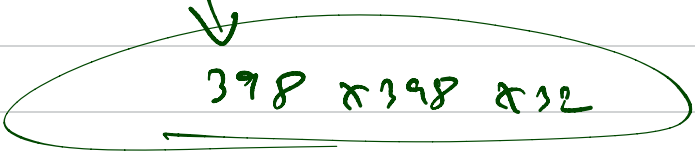








400 x 400 ⊙ 3 x 3 x 32



3 x 3 x 32 x 64