Crack the Captcha by CNN

RungChi

Agenda

- Mission - Crack the Captcha of TRA
- Process Flow
- Task 1 - Prepare Training Data for Model
  - First Try / Second Try
  - Hands On
- Task 2 - Build CNN Model
  - Concept Introduction
  - Hands On

Mission

Crack the Captcha of TRA

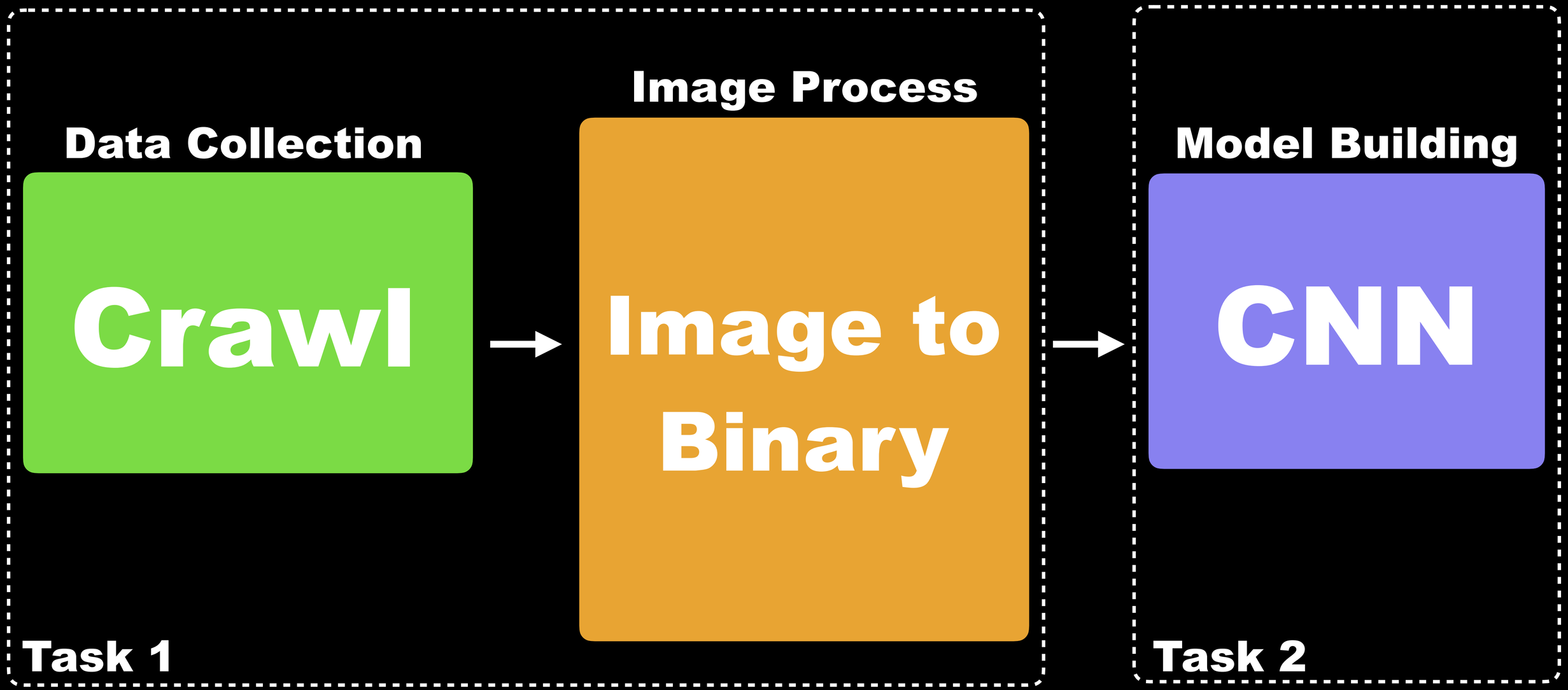
Mission to Crack TRA Captcha

→ 50378

→ 196974

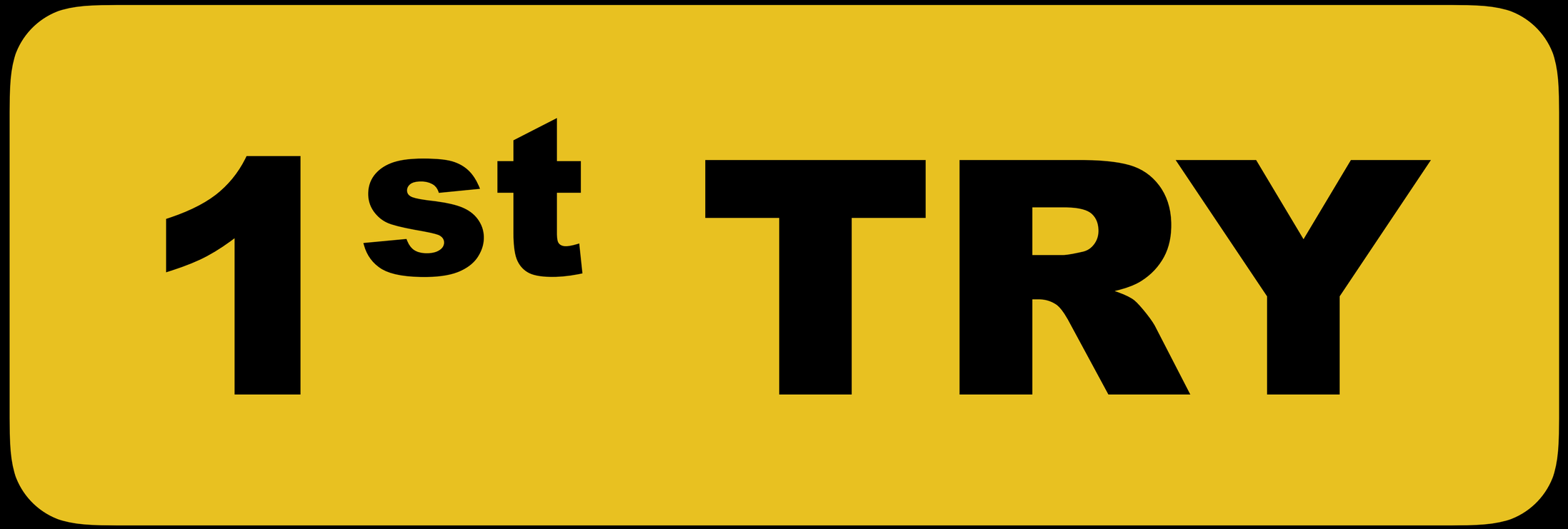
Process Flow

Process of Crack Captcha

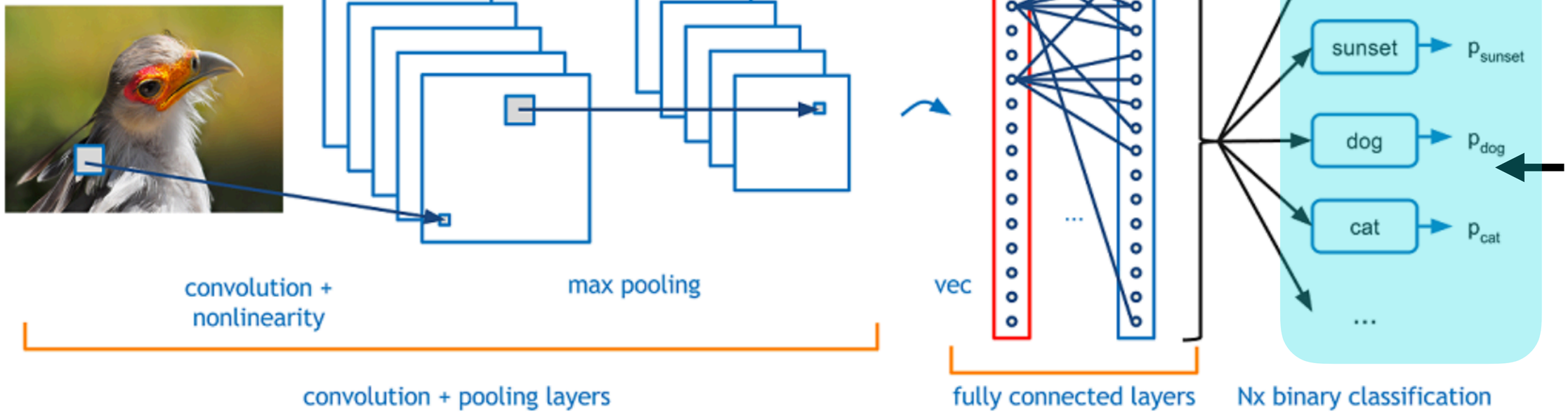


Task 1

Prepare Training Data for Model

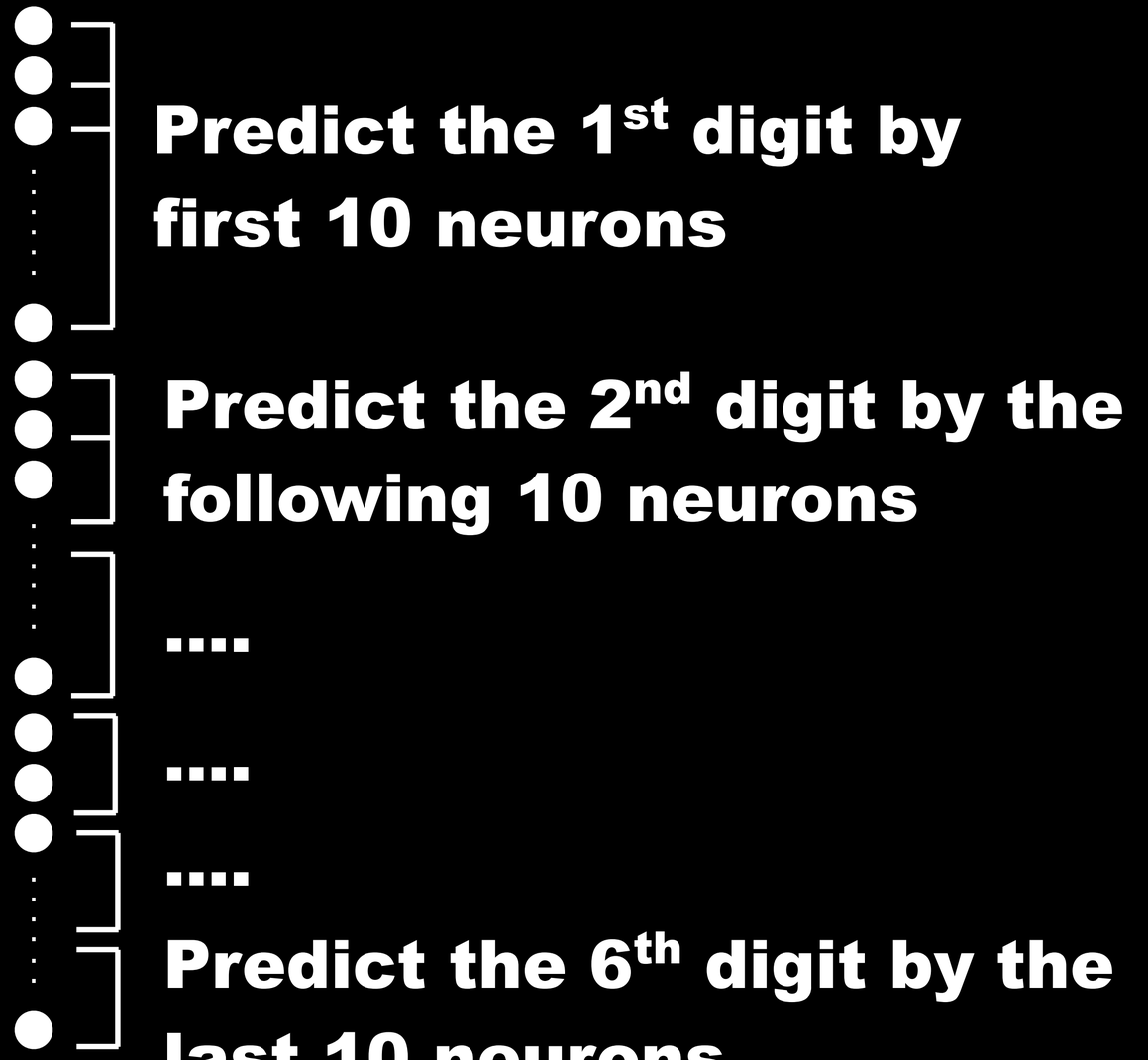

1st TRY

Thinking - CNN to classification



正確答案的類別

First Problem - how do we design the final layer?



First Problem - how do we design the final layer?

60 neurons

Second Problem - NOT Fixed number of digit

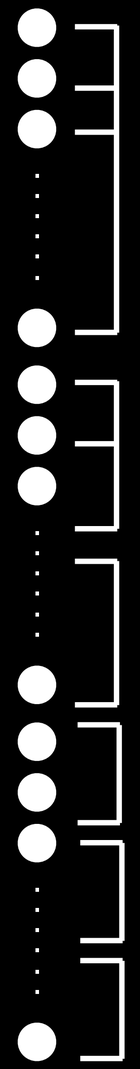
5 digits

6 digits

Second Problem - NOT Fixed number of digit

If the number of digit is 5, all of the last 10 neurons are 0 ...

Why the last 10 neurons?
Why not other neurons?



Predict the 1_{st} digit by first 10 neurons

Predict the 2_{nd} digit by the following 10 neurons

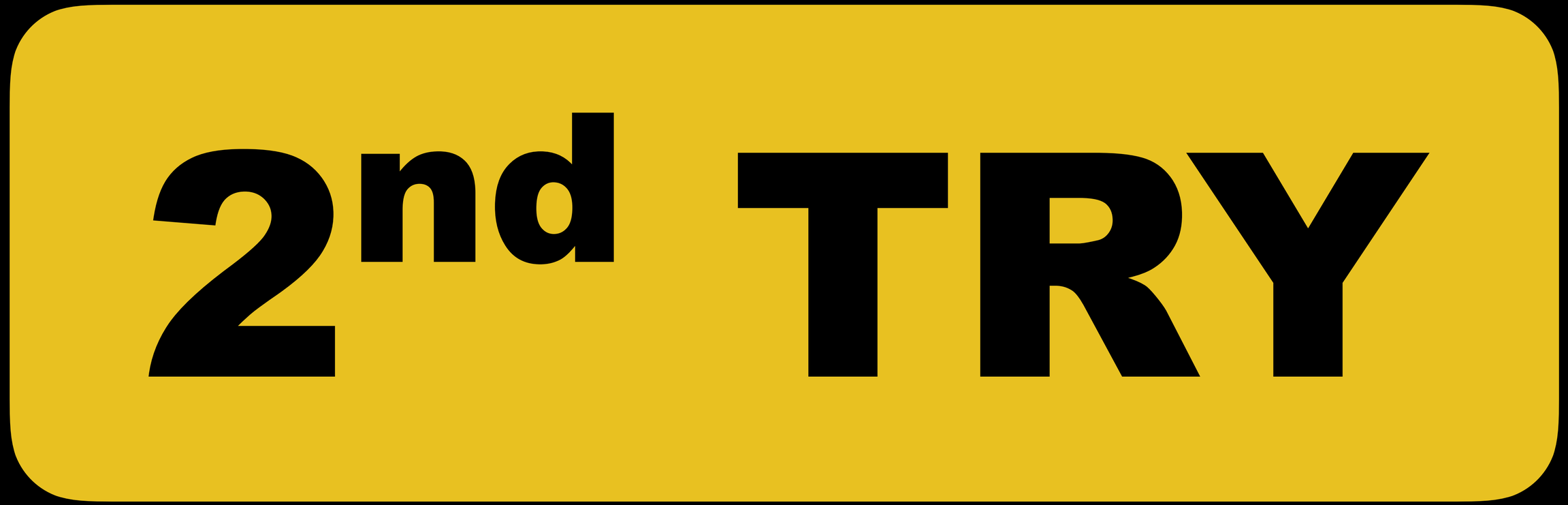
....

....

....

Predict the 6_{th} digit by the last 10 neurons

Give up by 10 mins


2nd TRY

MODEL

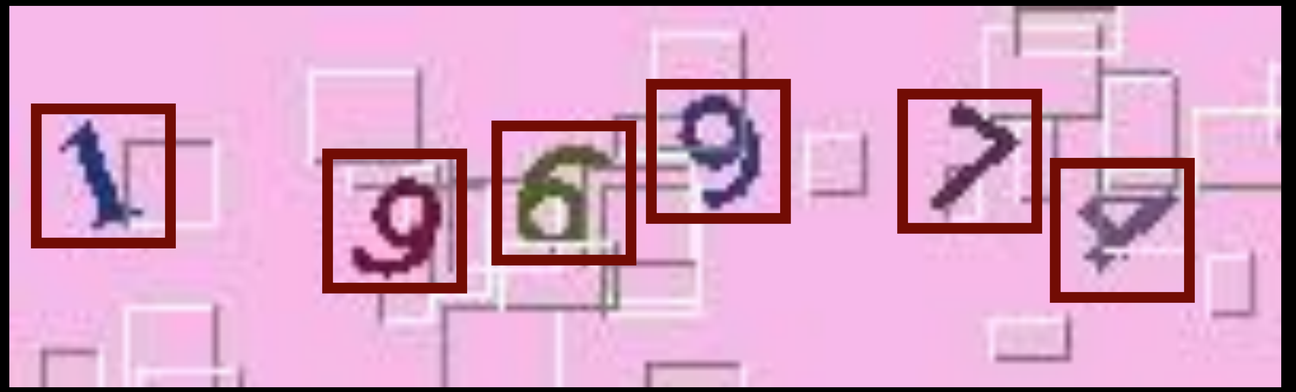
50378

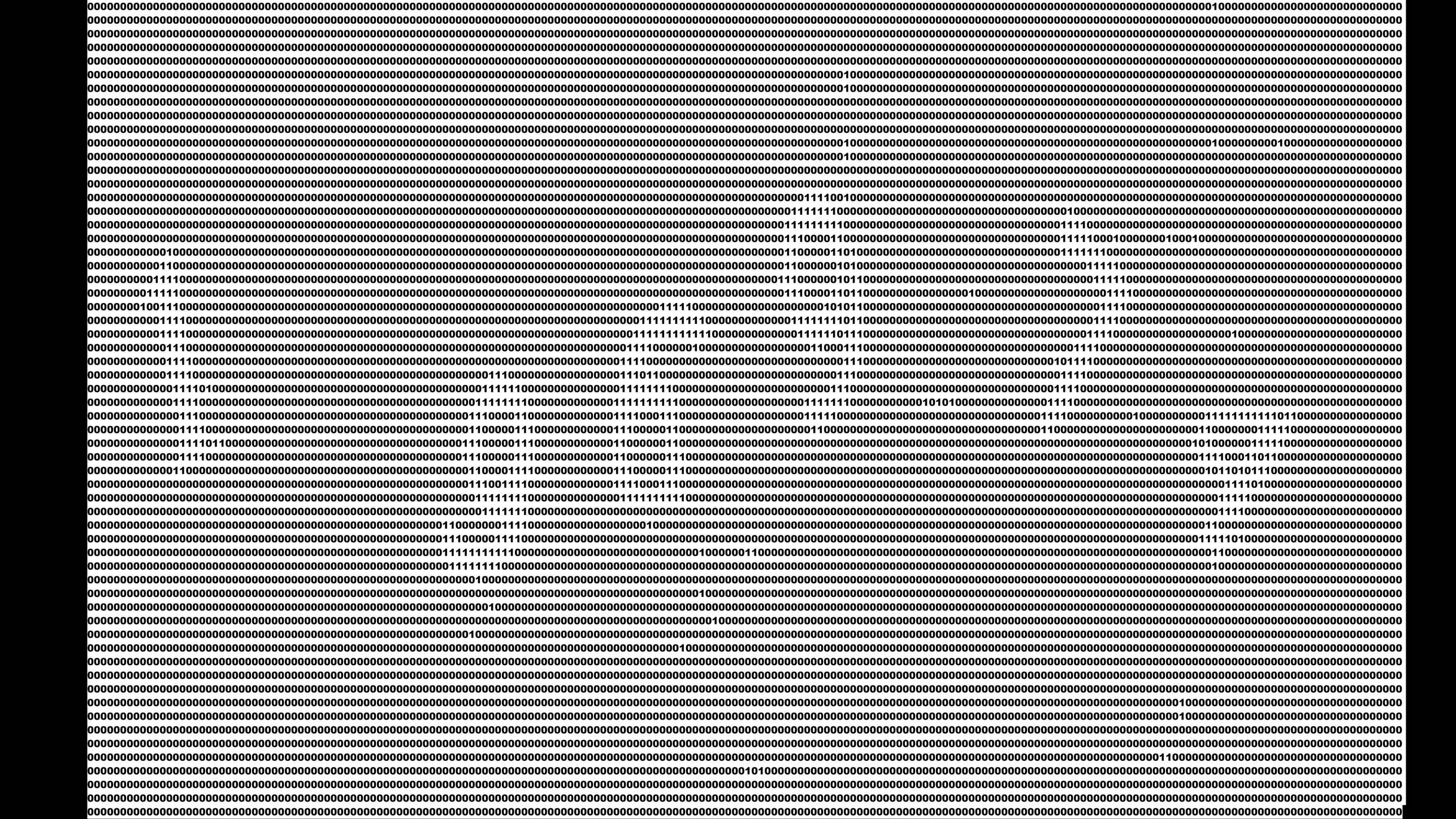
196974

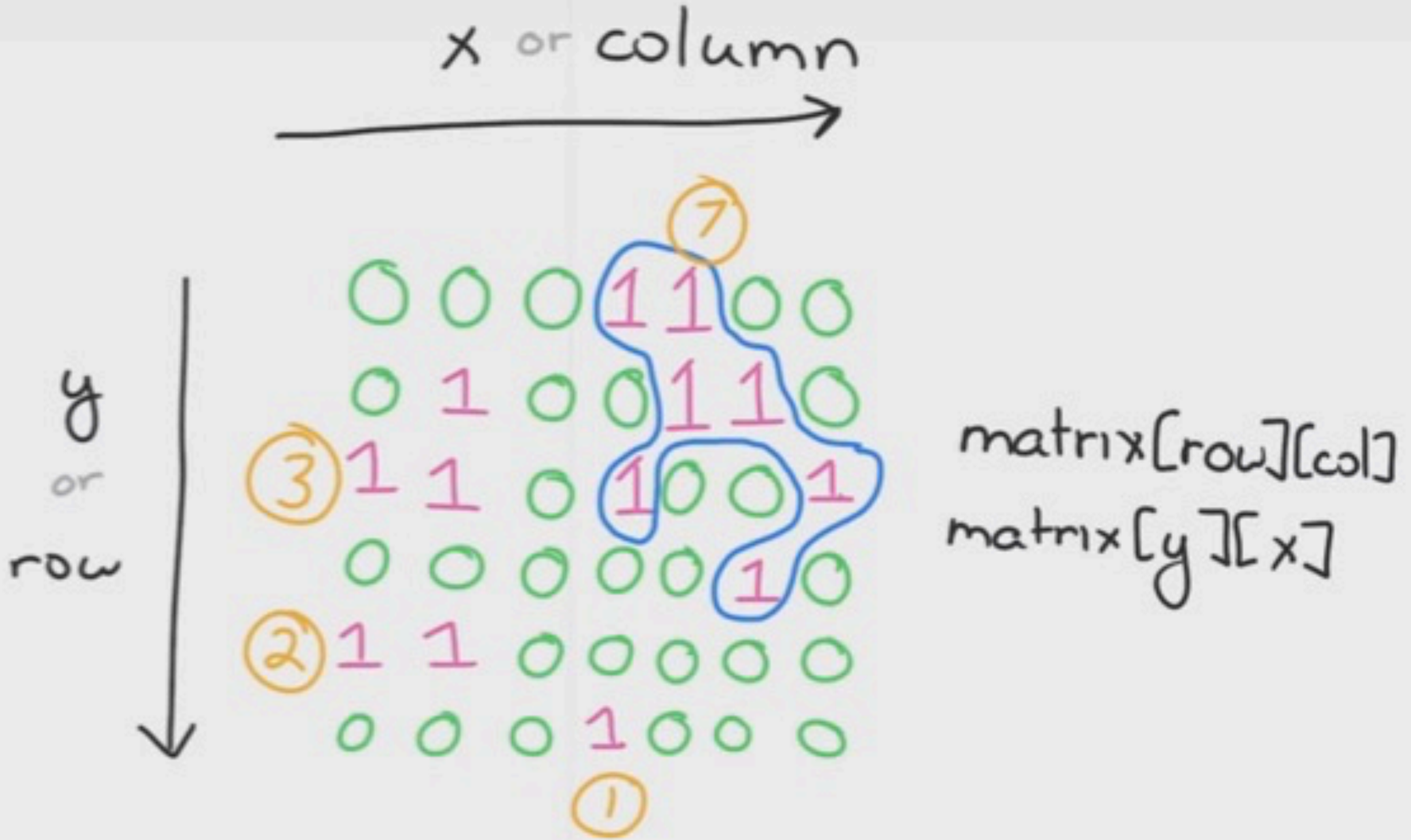
Another Problem - how to locate the ‘position’ of digit?

Locate the positions

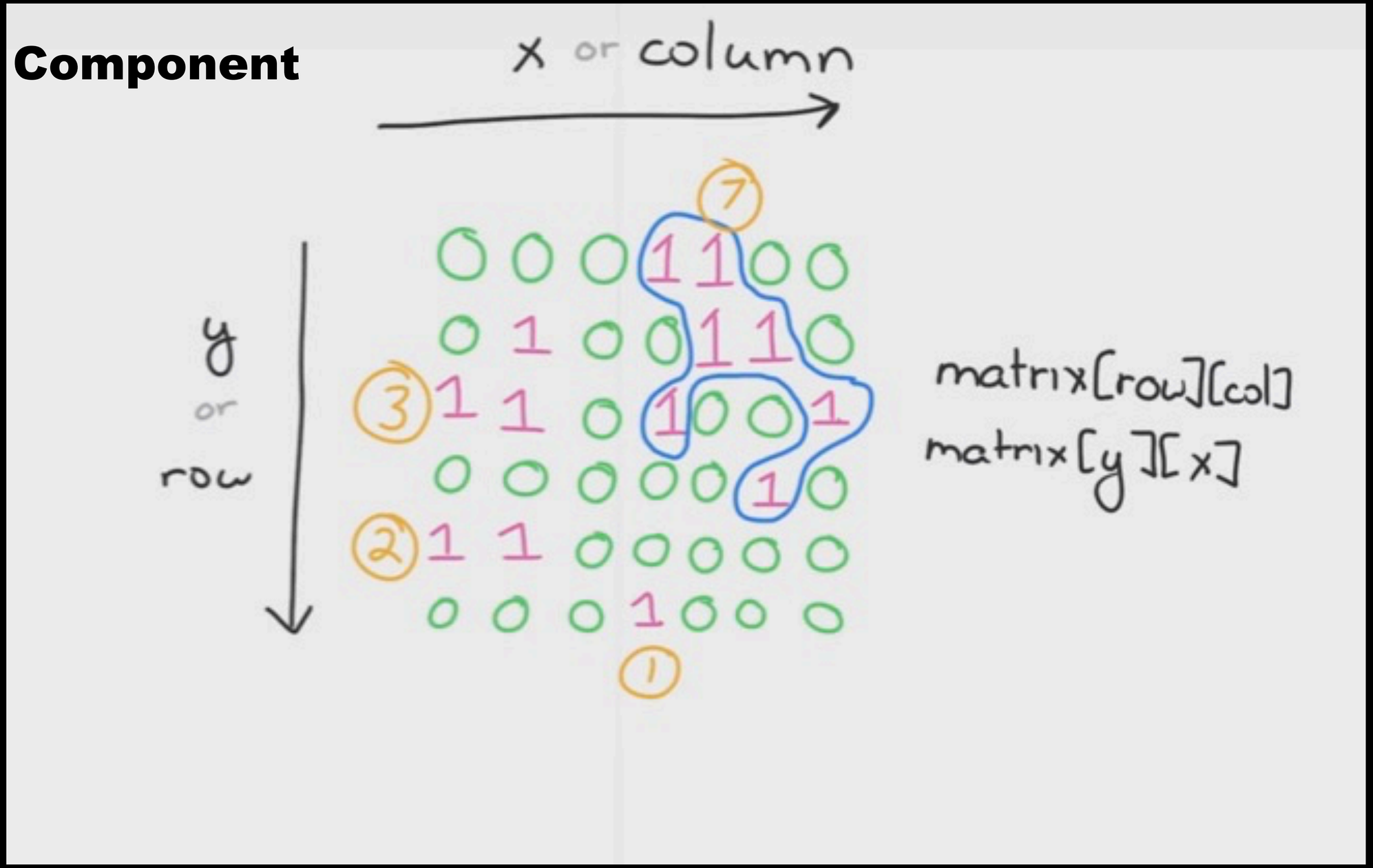
(right, top) and (left, bottom)






x or column
y
or
row
7
3
2
1
matrix[row][col]
matrix[y][x]

. Connected Component
  . DFS
  . BFS



Data Collection / Image Process

- Requirements
  - **pillow** - Python Image Library

Process of Crack Captcha

Data Collection

1a. Fetch Training Data

1b. Give the right answer for ever chatcha images

Image Process

2a. RGB to W/B style

2b. Locate ‘Number’ Position

2c. Cut off the digit from image

2d. Save ‘number’ image to numpy array

Model Building

?

Task 1

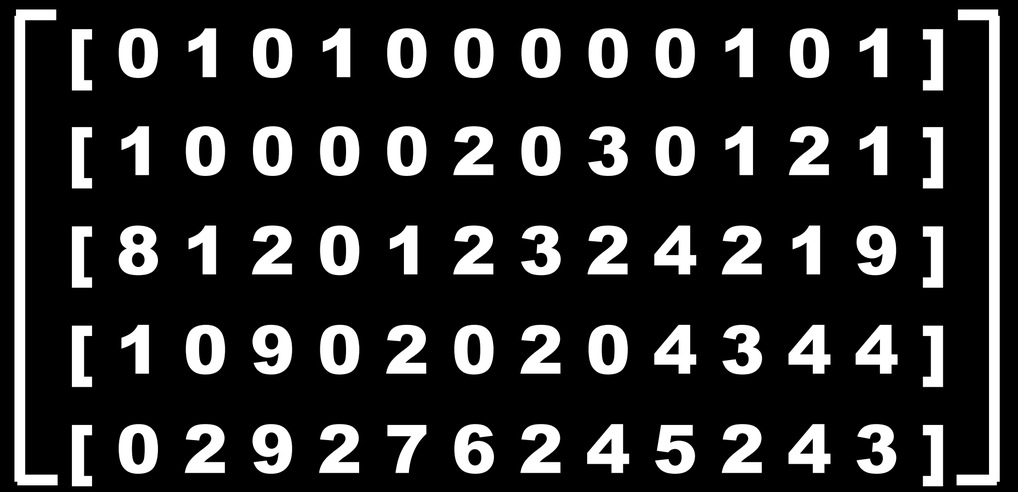
Task 2

Hands On!!

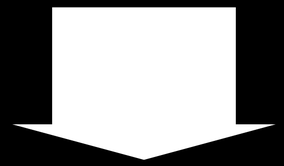
Task 2

Build CNN Model

Multi-Layer Perceptron



2D



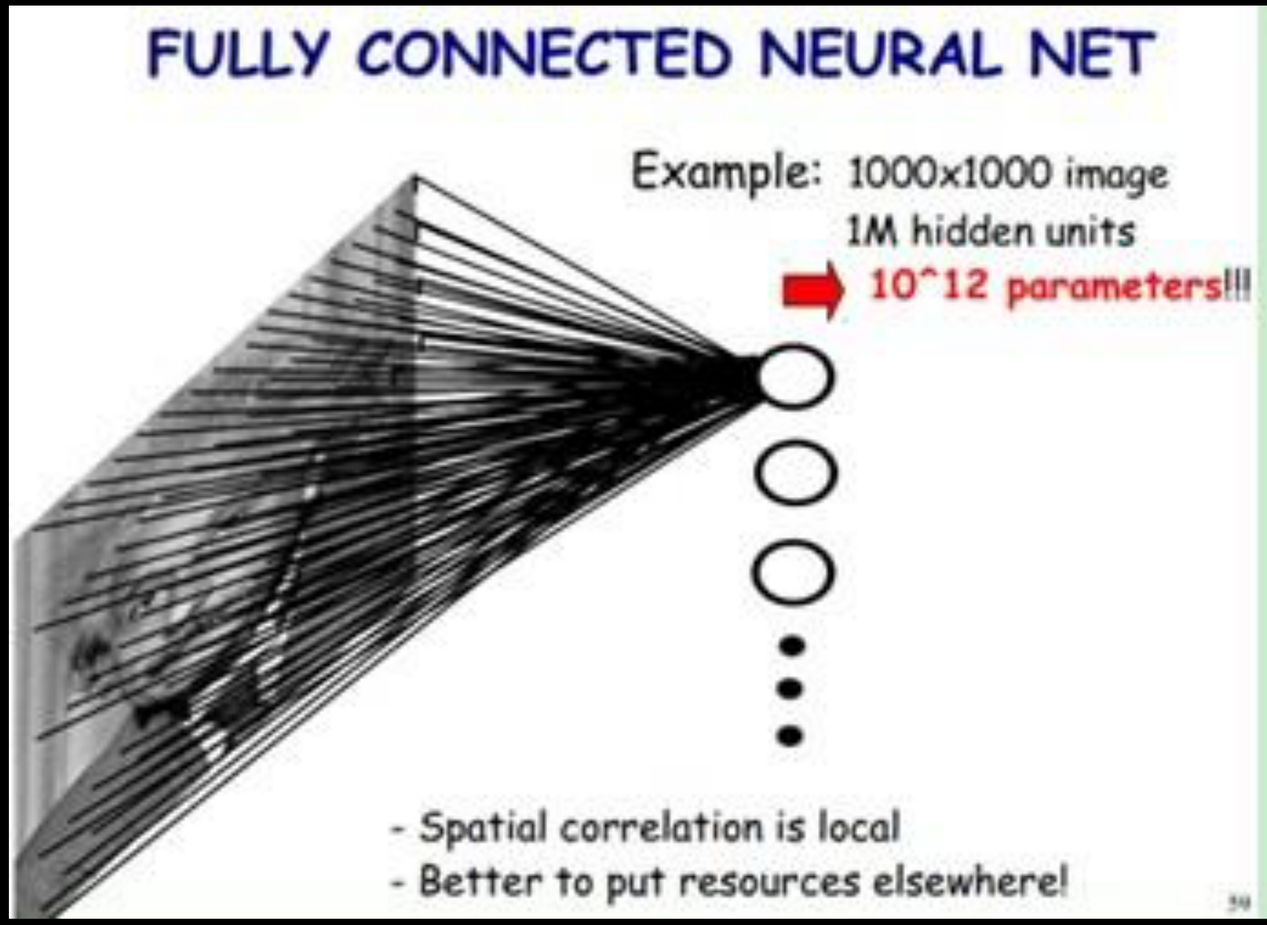
[1 0 1 0 0 0 0 0 1 0 1 1 0 0 0 0 2 0 3 0 1 2 1 8 1 2 0 1 2 3 2 4 2 1 9
1 0 9 0 2 0 2 0 4 3 4 4 0 2 9 2 7 6 2 4 5 2 4 3]

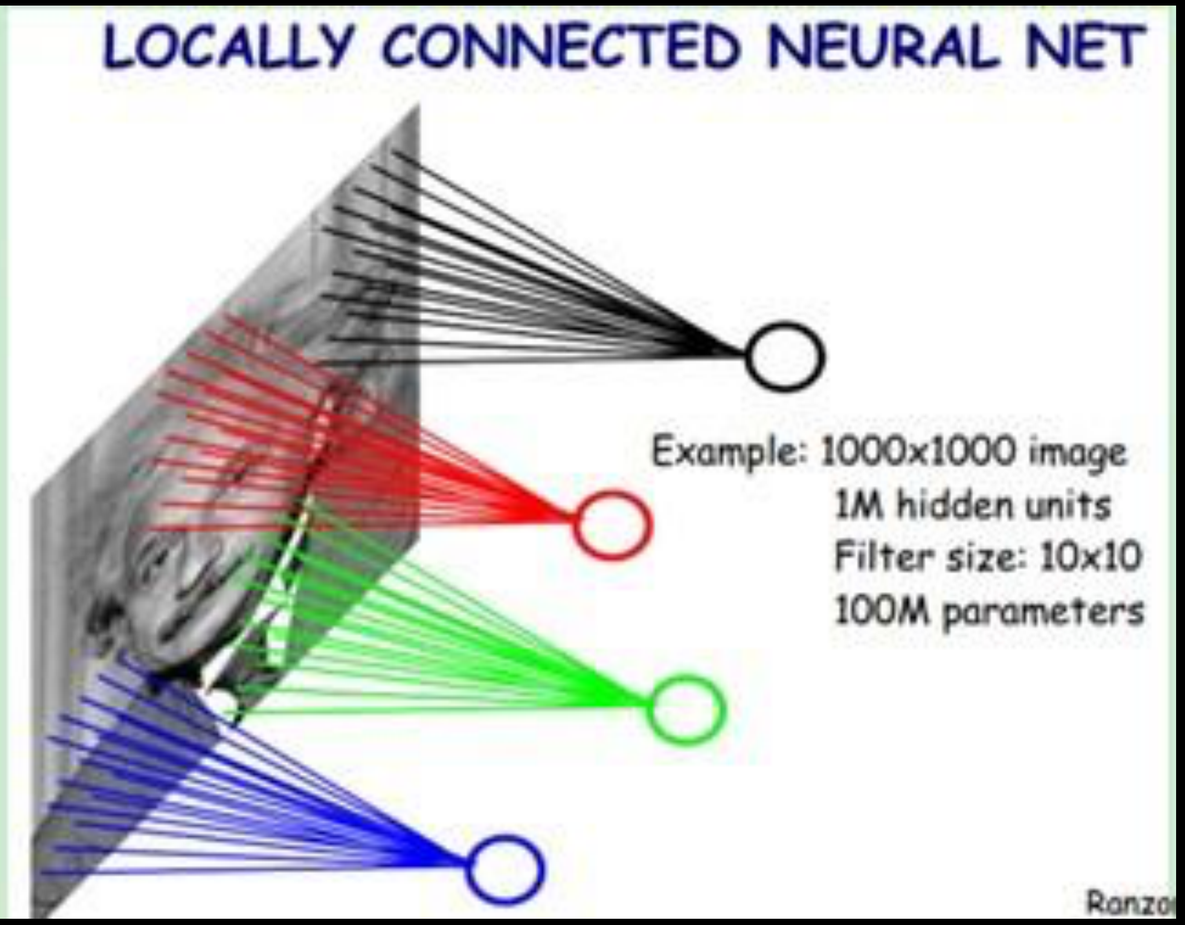
1D

CNN

- Convolutional Neural Networks
  1. Locally Connected vs. Fully Connected
  2. Convolutions
  3. Pooling

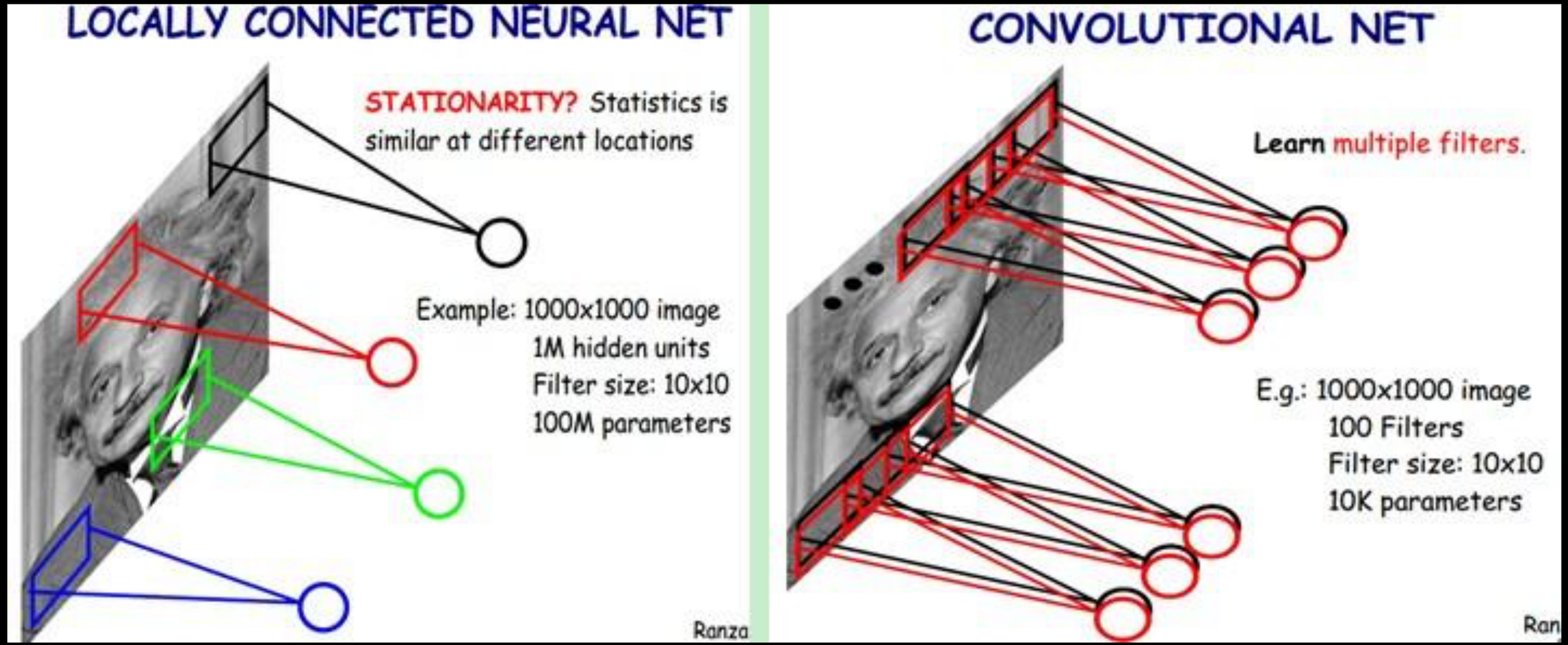
Convolutional Neural Networks





Locally Connected vs. Fully Connected

Convolutional Neural Networks

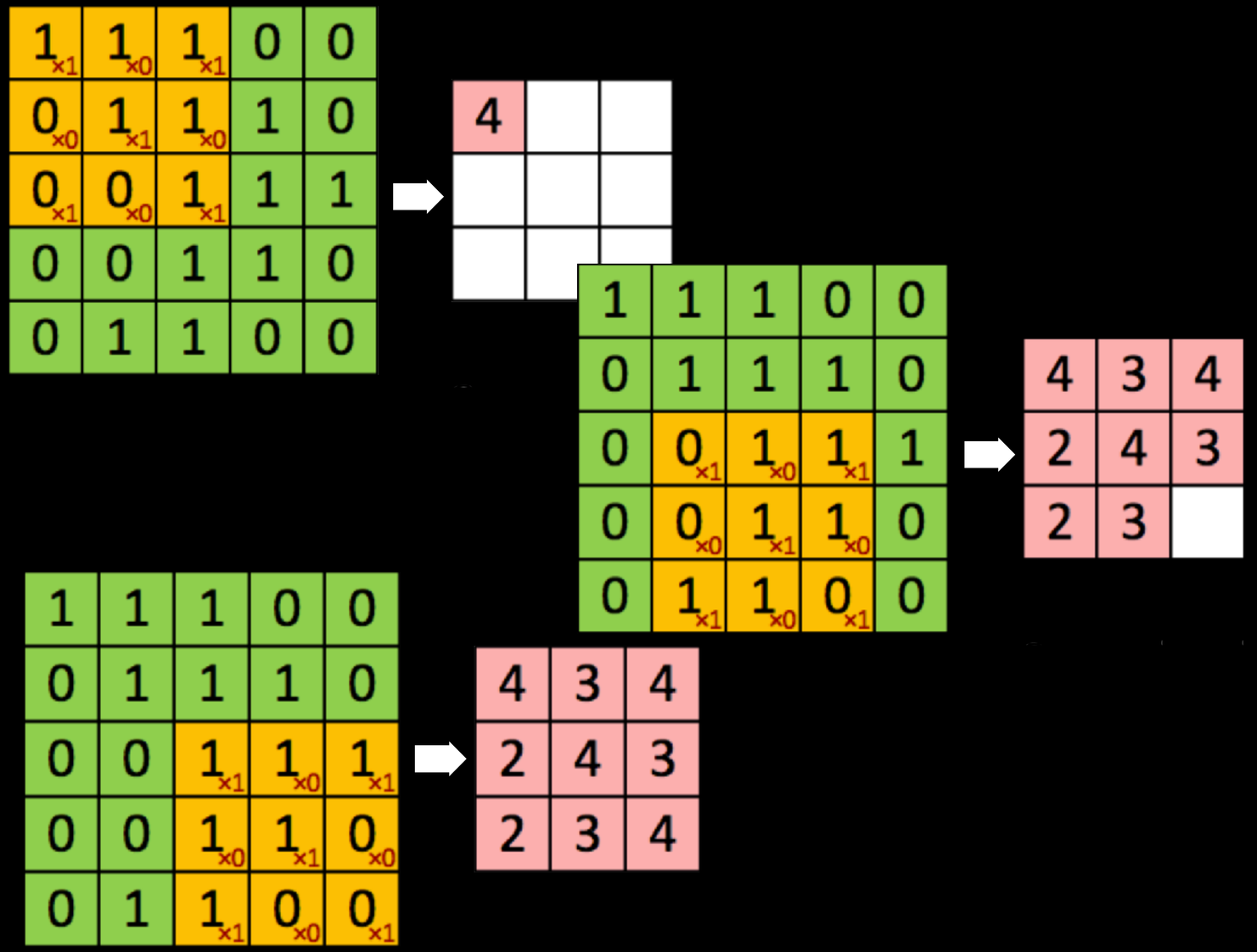


Convolutions

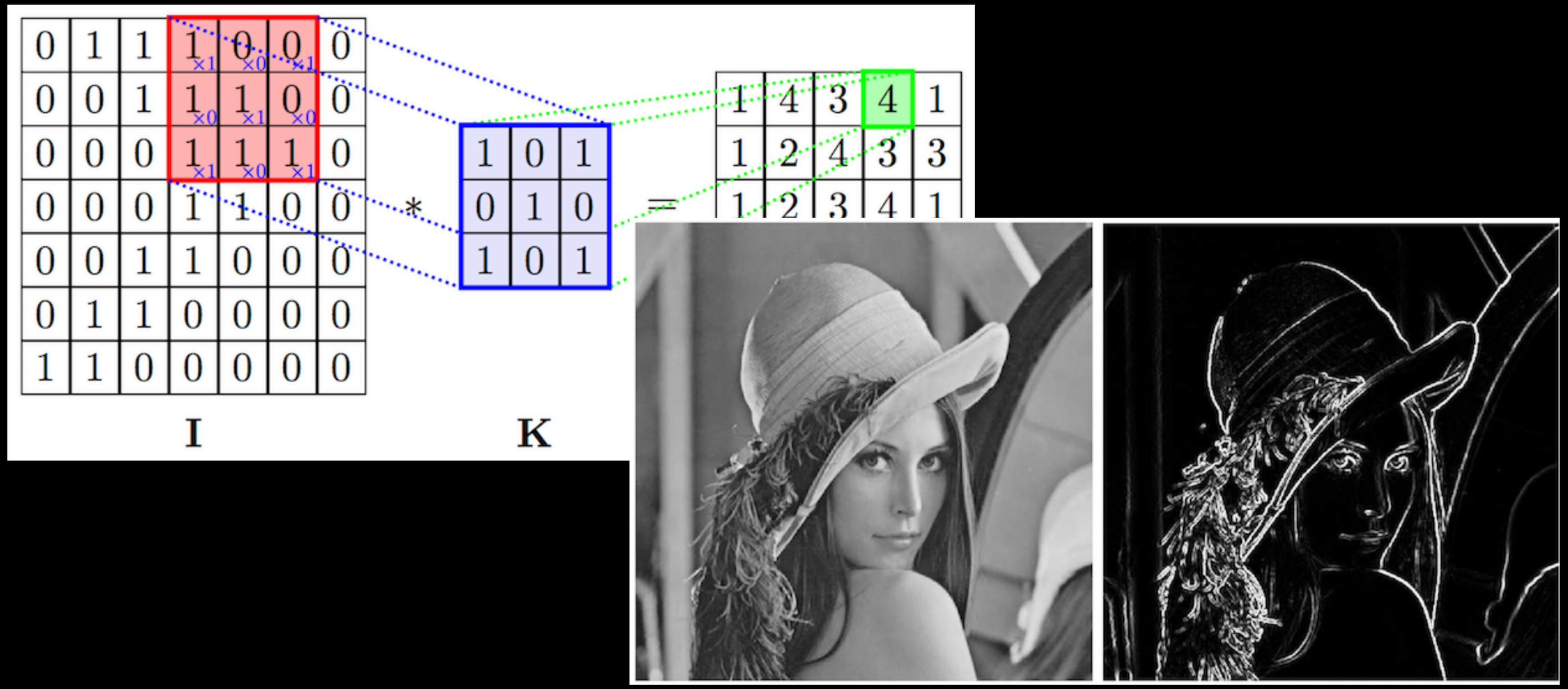
Convolutional Features

Filter

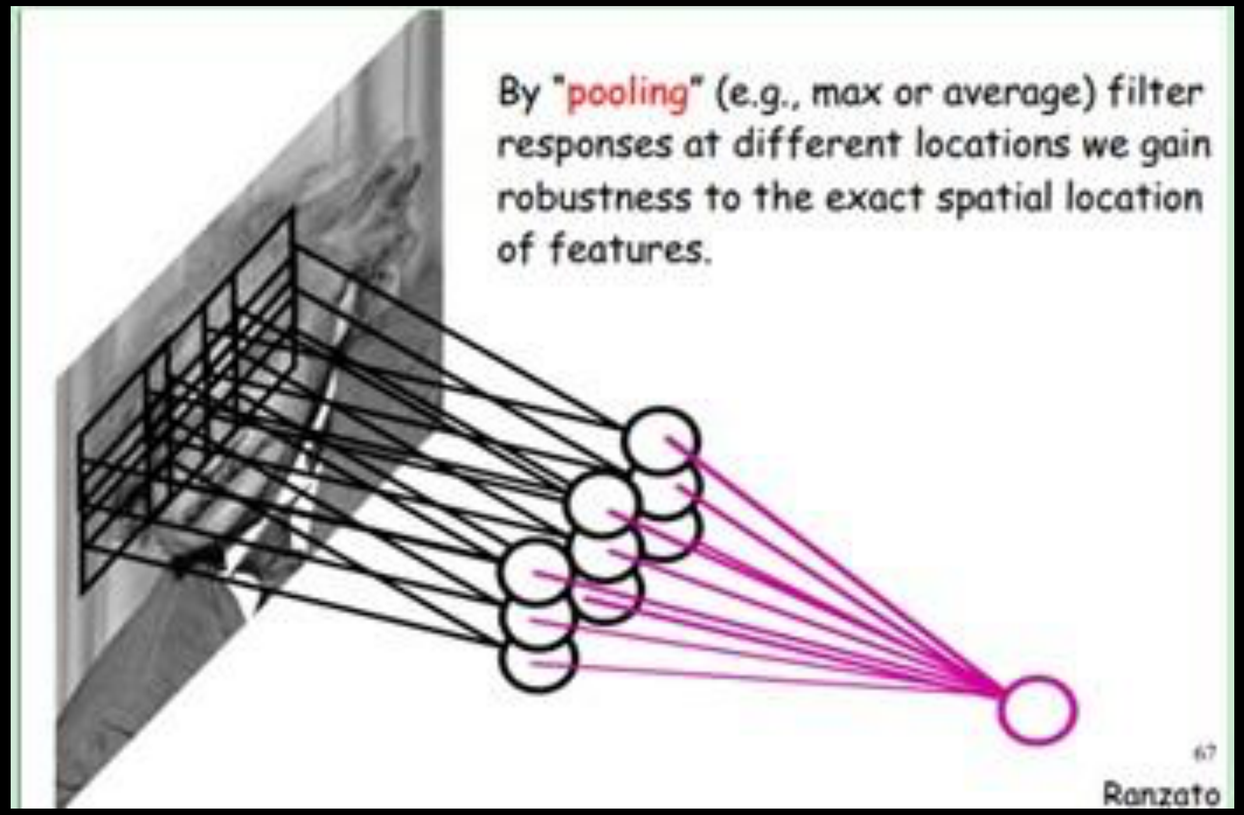
$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

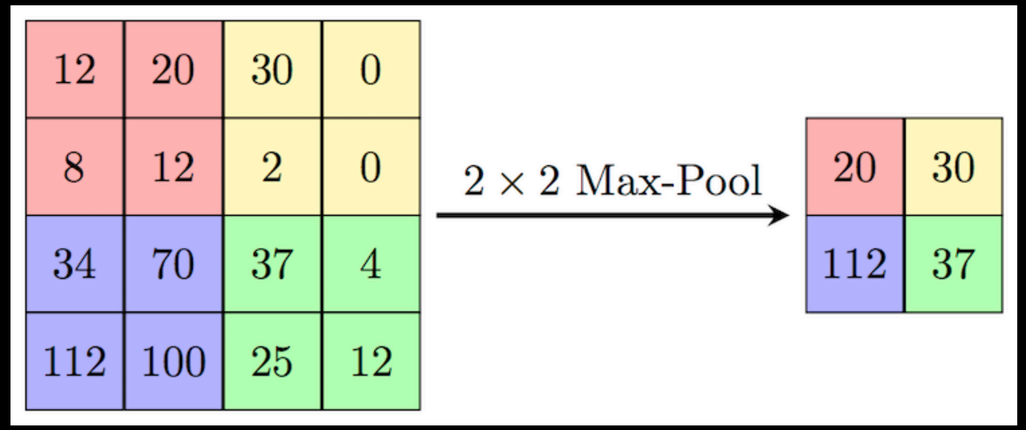


Convolutional Features

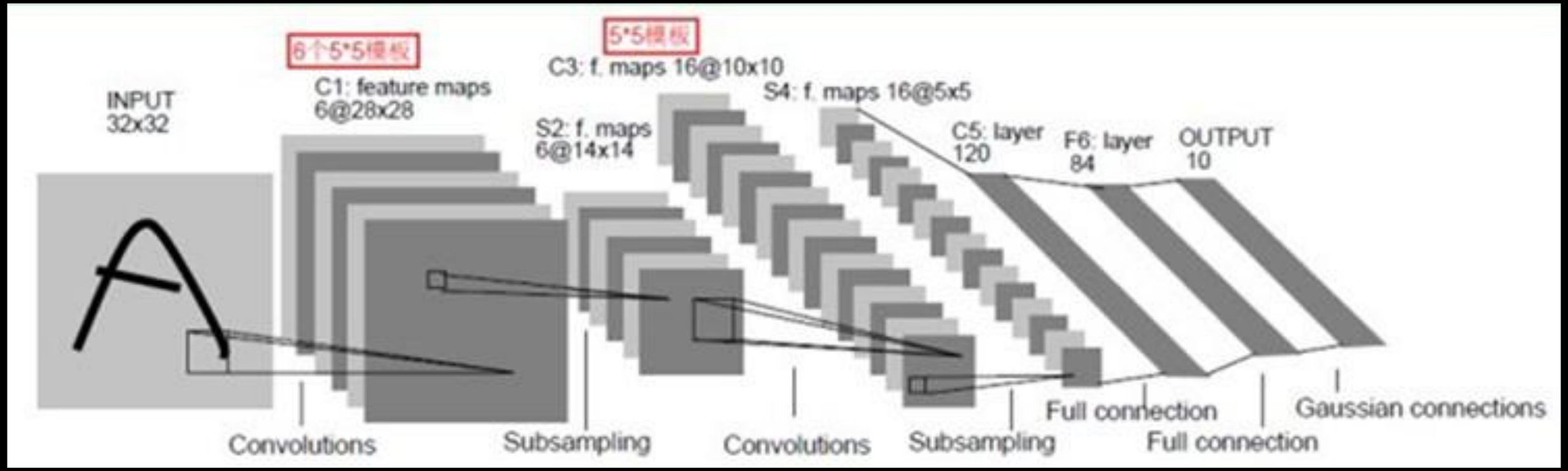


Pooling


By "pooling" (e.g., max or average) filter responses at different locations we gain robustness to the exact spatial location of features.
67
Ranzato


12 20 30 0
8 12 2 0
34 70 37 4
112 100 25 12
2 × 2 Max-Pool
20 30
112 37

Convolutional Neural Networks

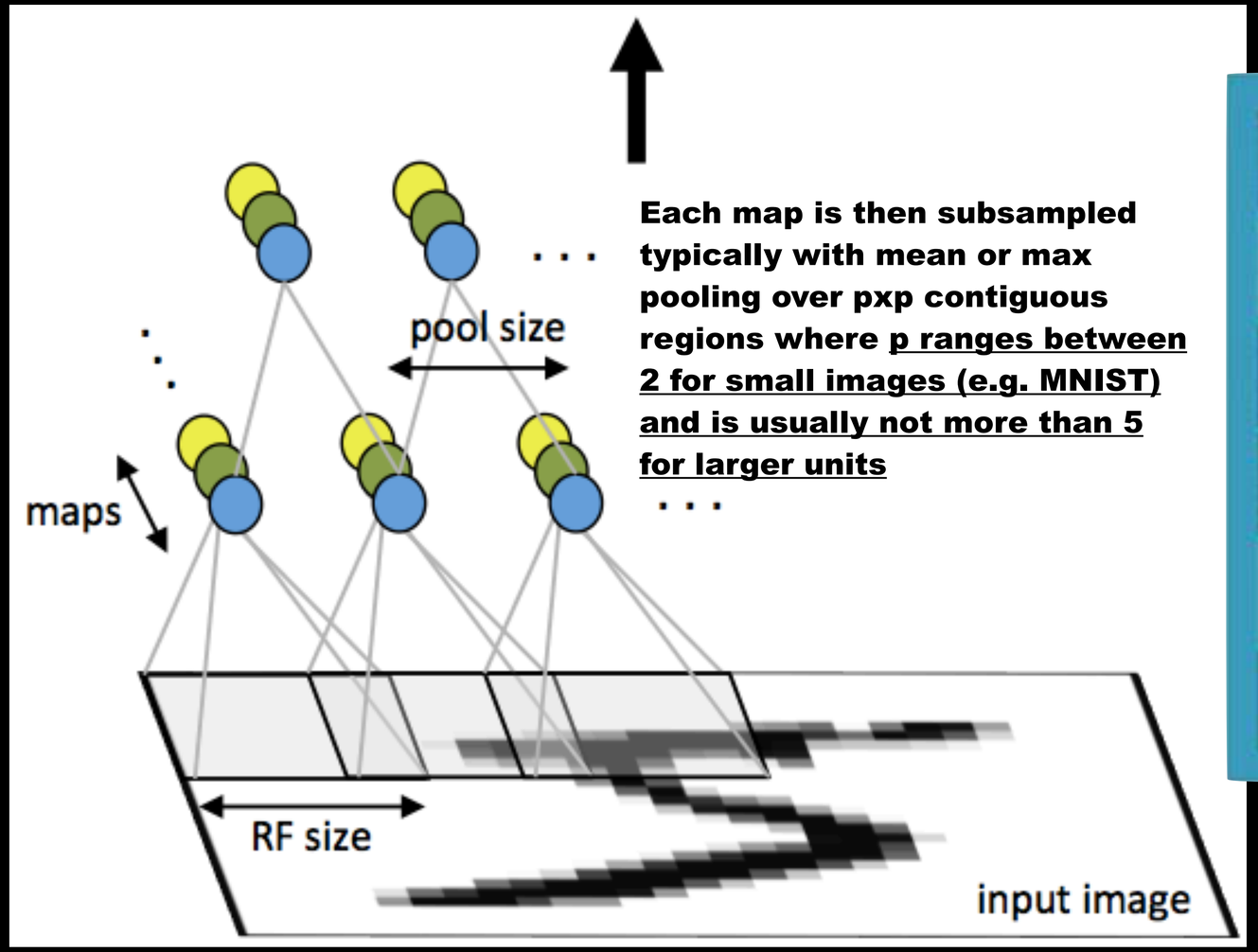


Convolutional Neural Networks

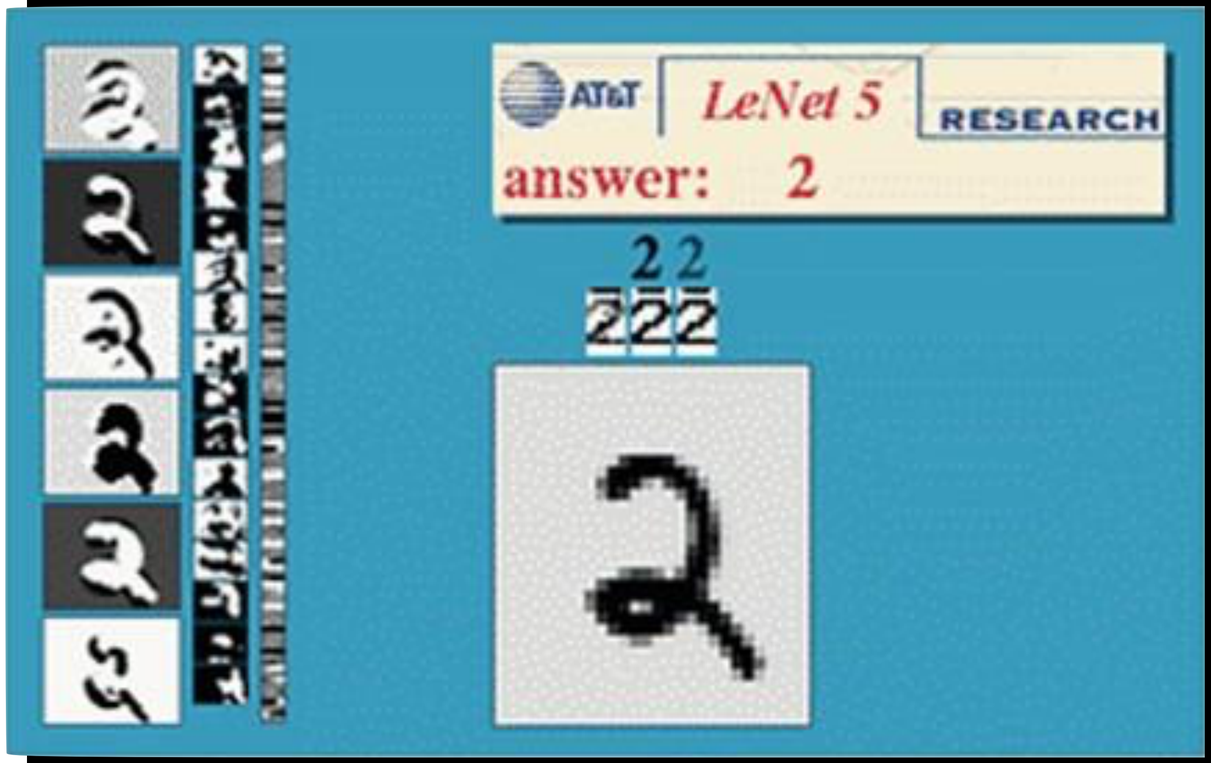
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	X				X	X	X			X	X	X	X		X	X
1	X	X				X	X	X			X	X	X	X		X
2	X	X	X				X	X	X			X		X	X	X
3		X	X	X			X	X	X	X			X		X	X
4			X	X	X			X	X	X	X		X	X		X
5				X	X	X			X	X	X	X		X	X	X

S2 -> C3

Convolutional Neural Networks



Each map is then subsampled typically with mean or max pooling over pxp contiguous regions where p ranges between 2 for small images (e.g. MNIST) and is usually not more than 5 for larger units



How convolutional neural networks see the world

CNN Model

- **Requirements**
  - **numpy** - transfer the image to the binary data
  - **tensorflow**, **keras** - CNN model

Process of Crack Captcha

Data Collection

1a. Fetch Training Data

1b. Give the right answer for ever chatcha images

Image Process

2a. RGB to W/B style

2b. Locate 'Number' Position

2c. Cut off the digit from image

2d. Save 'number' image to numpy array

Model Building

3a. MLP Model

3b. 1-Layer CNN

3c. 2-Layer CNN

Task 1

Task 2

Hands On!!