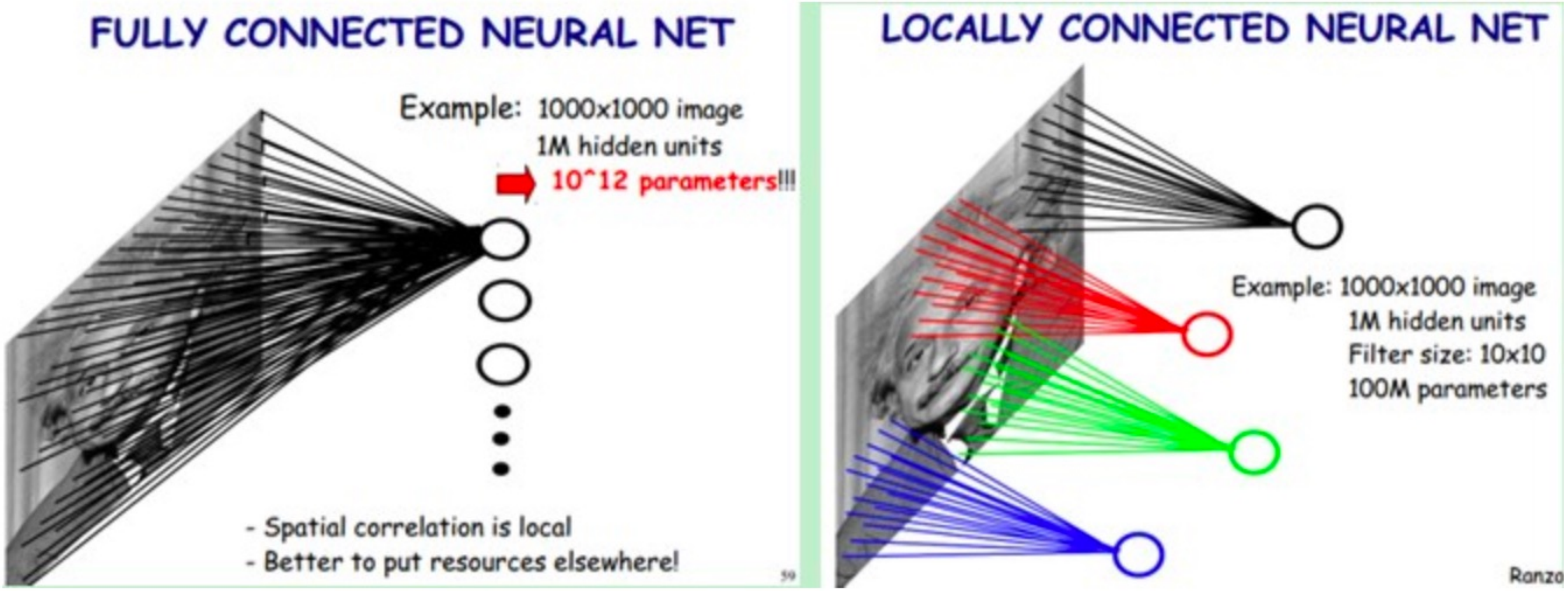
Convolutional Neural Network

Spring 2024

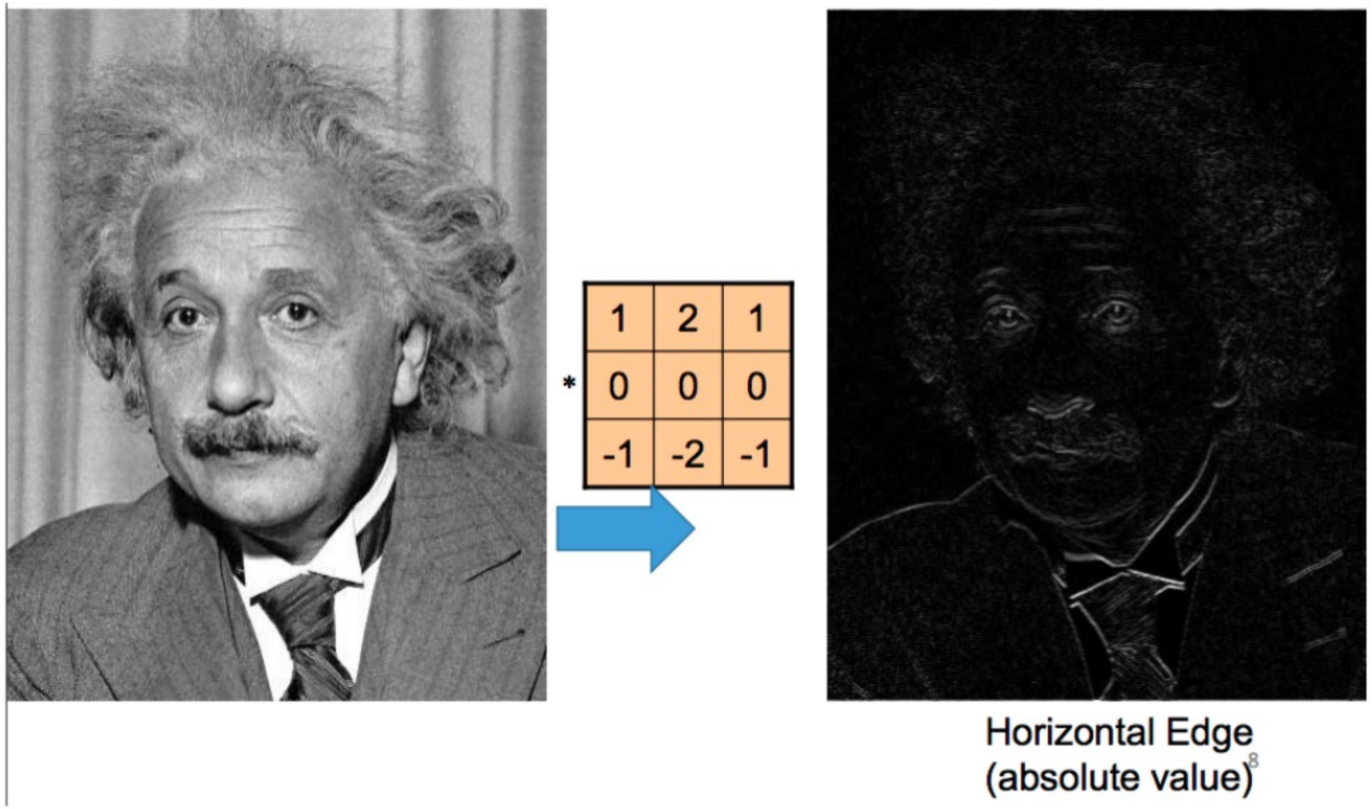
Hongchang Gao

Background


FULLY CONNECTED NEURAL NET
Example: 1000x1000 image
1M hidden units
10^12 parameters!!!
- Spatial correlation is local
- Better to put resources elsewhere!
LOCALLY CONNECTED NEURAL NET
Example: 1000x1000 image
1M hidden units
Filter size: 10x10
100M parameters
Ranzo

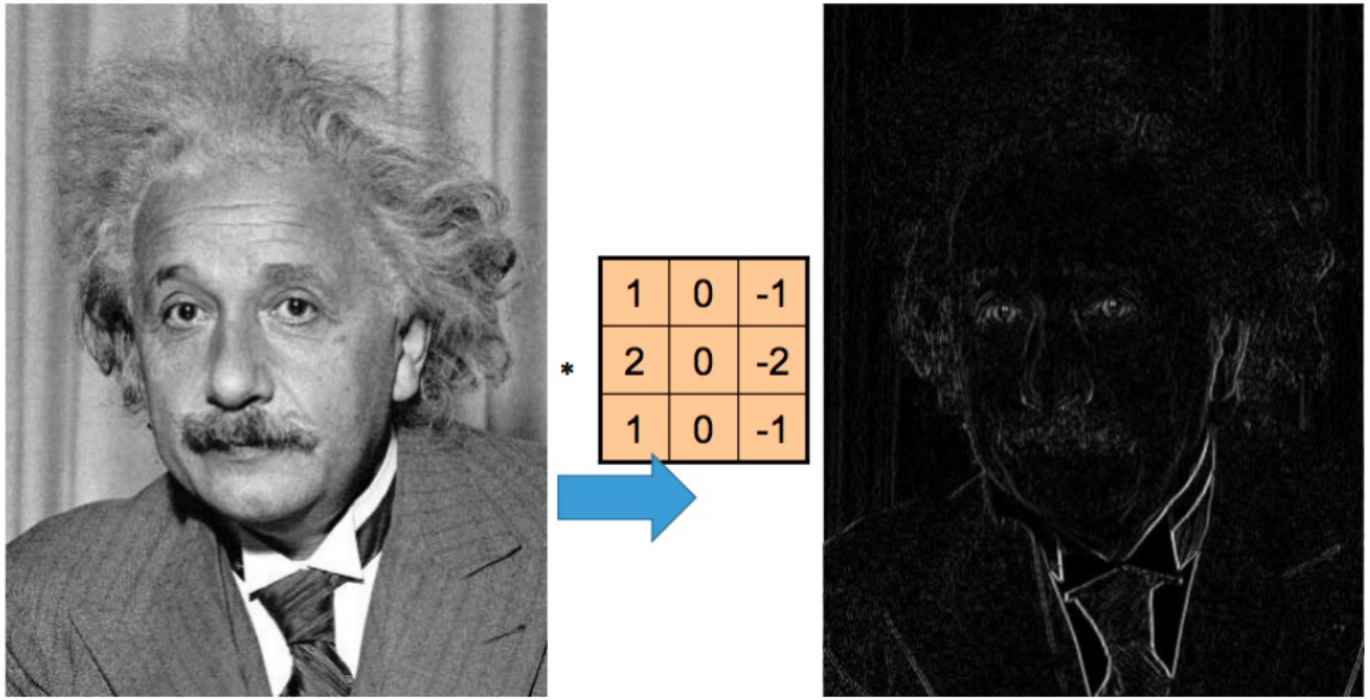
Background

- Feature extractor: Learn useful features from images for prediction
  - 1. Sobel Filter - Weights to Detect Horizontal Edges



Background

- Feature extractor: Learn useful features from images for prediction
    - 1. Sobel Filter - Weights to Detect Vertical Edges



Vertical Edge
(absolute value)

Convolution

- Matrix Inner Product

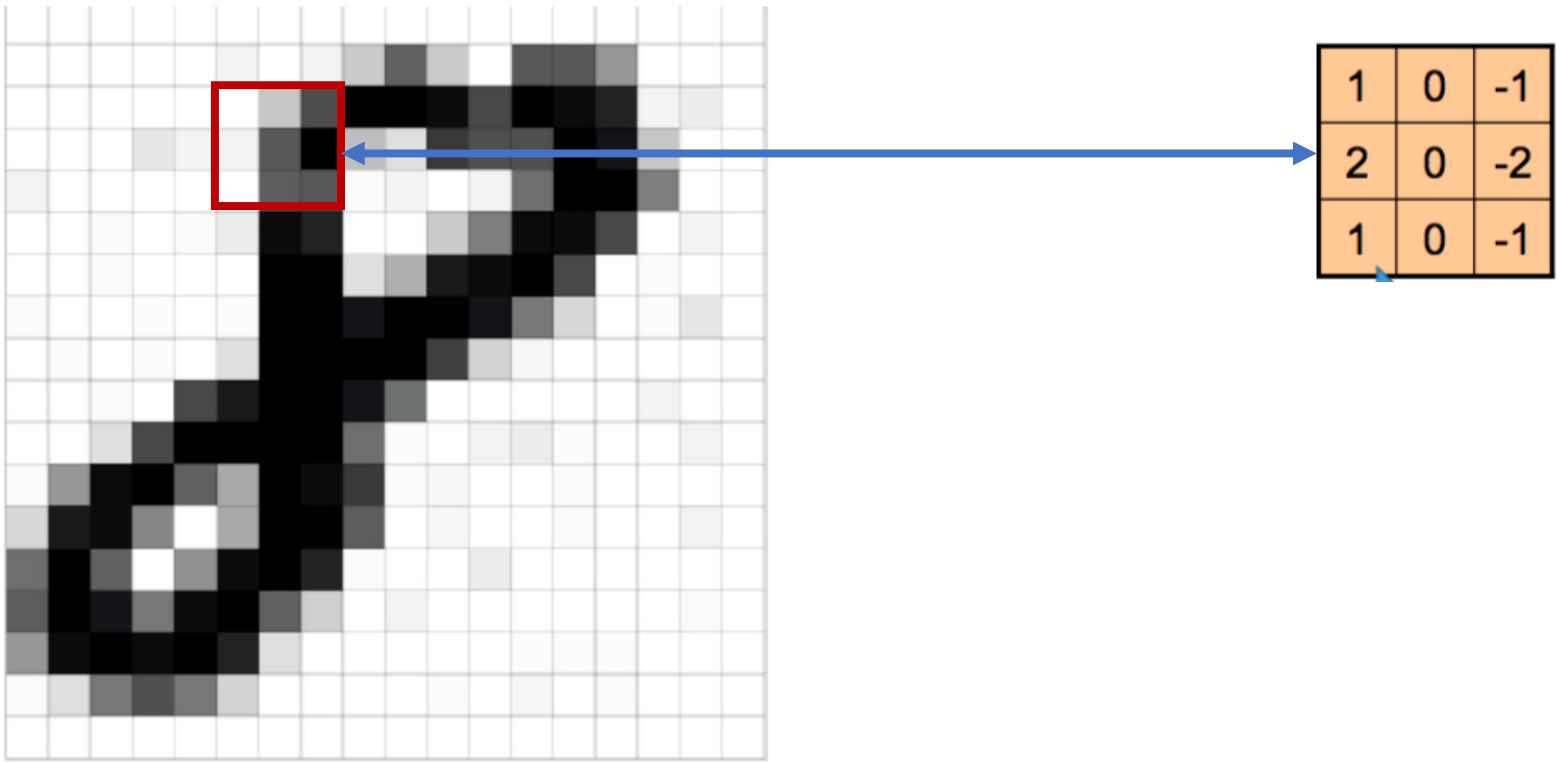
  - $\mathbf{A} = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$ and $\mathbf{B} = \begin{bmatrix} 5 & 7 \\ 6 & 8 \end{bmatrix}$.

  - Inner product:

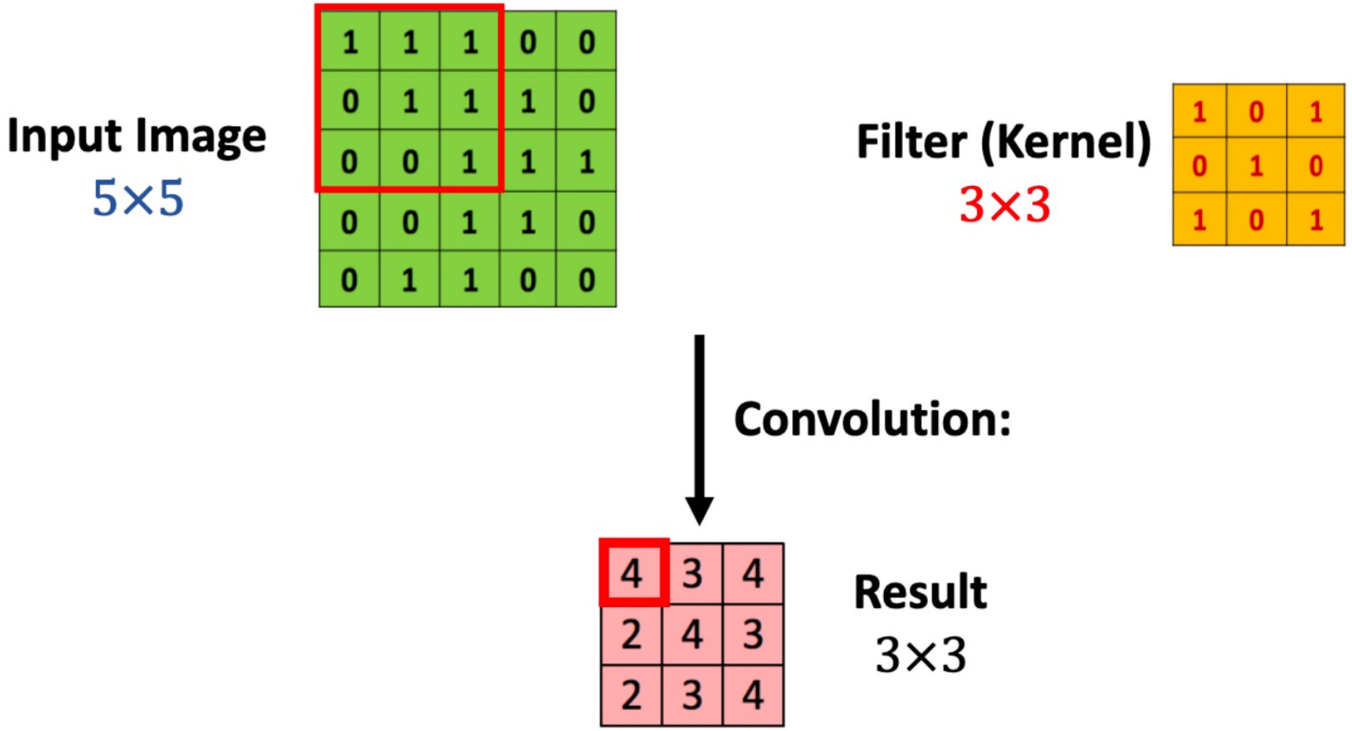
$$\langle \mathbf{A}, \mathbf{B} \rangle = \sum_i \sum_j a_{ij} b_{ij} = 70.$$

  - Property: $\langle \mathbf{A}, \mathbf{B} \rangle = \langle \mathrm{vec}(\mathbf{A}), \mathrm{vec}(\mathbf{B}) \rangle$.

Convolution

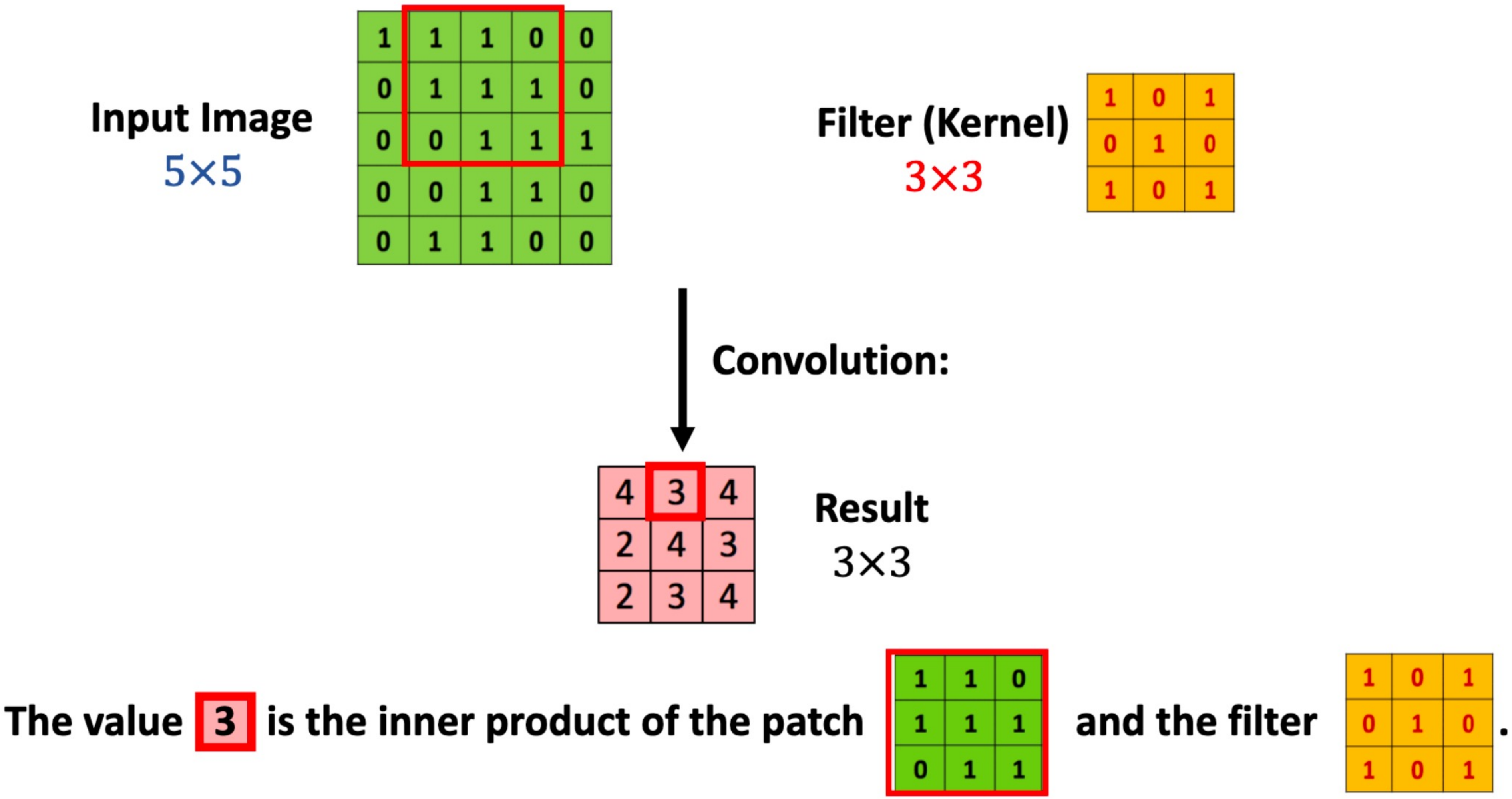


Convolution

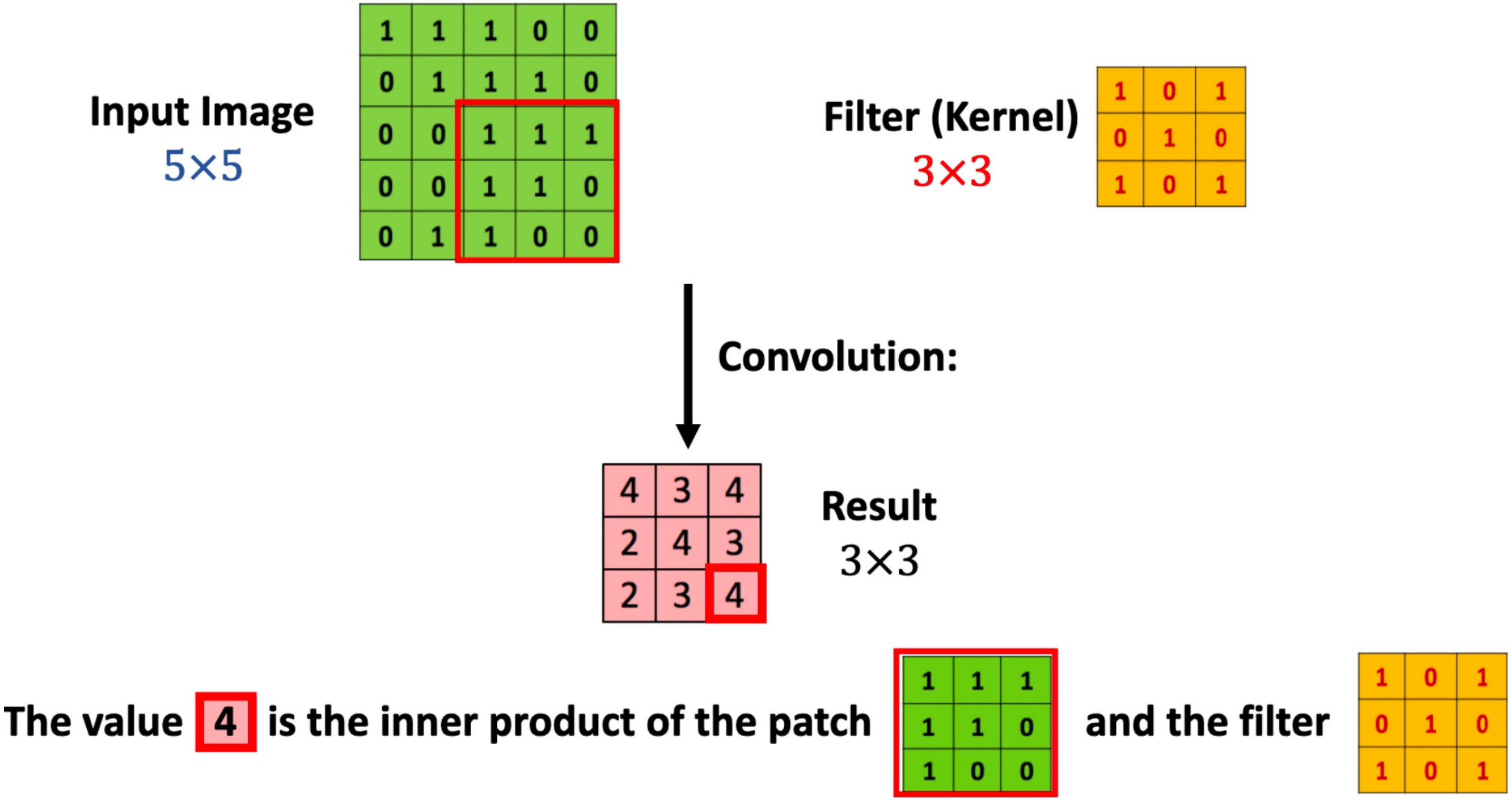


The value $\boxed{4}$ is the inner product of the patch $\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$ and the filter $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$.

Convolution



Convolution



Convolution

- Question: How many 3x3 patches in following two images?

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

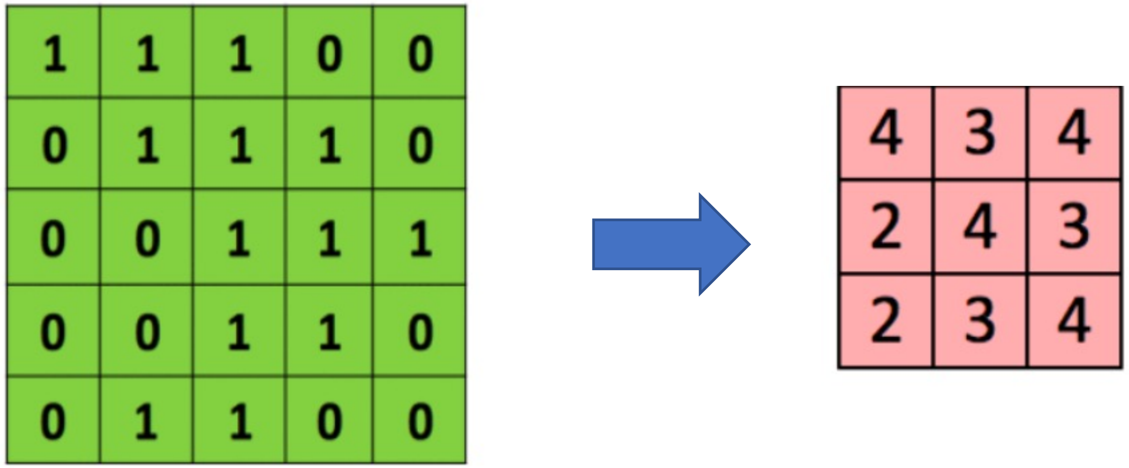
1	1	1	0
0	1	1	1
0	1	1	0
1	1	0	0

Dimensions:

- Input: $d_1 \times d_2$
- Filter: $k_1 \times k_2$
- Output: $(d_1 - k_1 + 1) \times (d_2 - k_2 + 1)$

Zero Padding

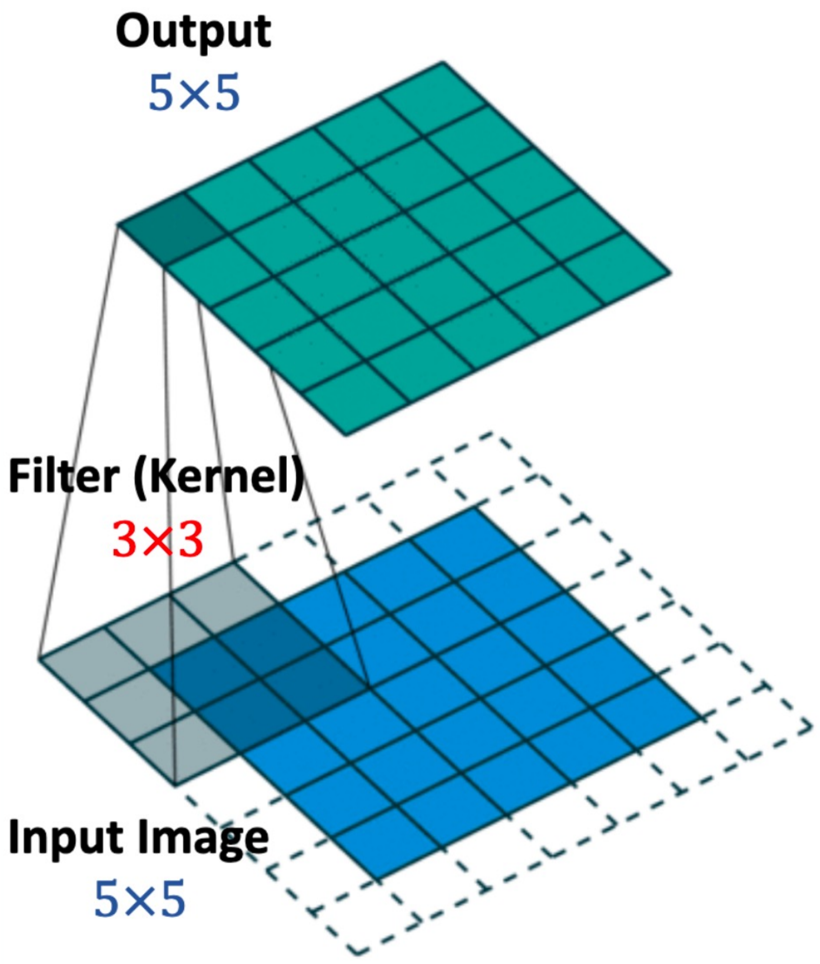
- Problem: the output is smaller than the input



- Zero padding
  - Keep the next layer's width and height consistent with the previous
  - Keep the information around the border of the image

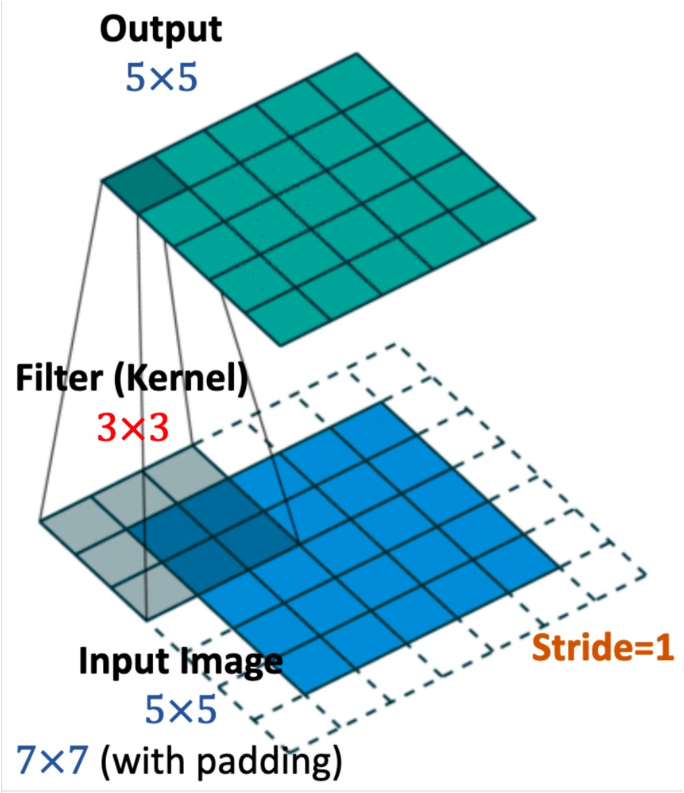
Zero Padding

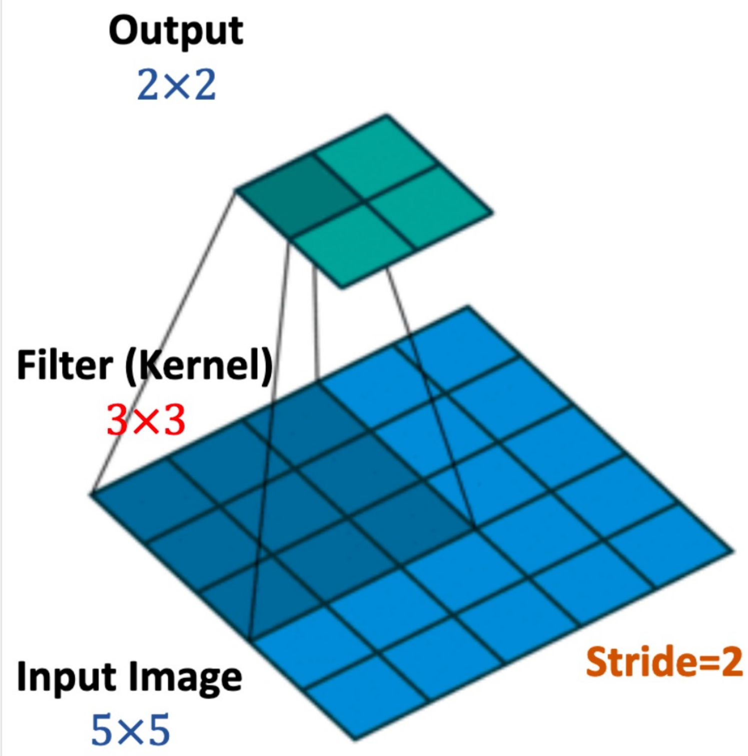
- Add a “boarder” of all-zeros.
- Increase the input shape:
  - From $d_1 \times d_2$ to $(d_1 + 2) \times (d_2 + 2)$.
- If the filter is 3×3, the output is $d_1 \times d_2$



Stride

- Stride
  - The filter moves K step each time, K >=1





Stride

- Stride
  - The filter moves K step each time, K >=1

Dimensions:

- Input: $d_1 \times d_2$
- Filter: $k_1 \times k_2$
- Stride: s
- Output: $\left(\left\lfloor \frac{d_1 - k_1}{s} \right\rfloor + 1\right) \times \left(\left\lfloor \frac{d_2 - k_2}{s} \right\rfloor + 1\right)$

Summary

- Convolution:
  - Feature map, filter/kernel
  - Zero padding
  - Stride

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

Feature map

1	0	1
0	1	0
1	0	1

Filter/kernel

4	3	4
2	4	3
2	3	4

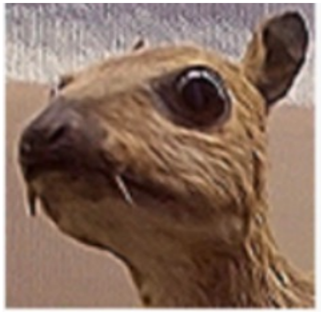
Feature map

Summary

- Traditional Feature extractor:
  - hand-crafted filters
  - a lot of expertise
  - Limited filters

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

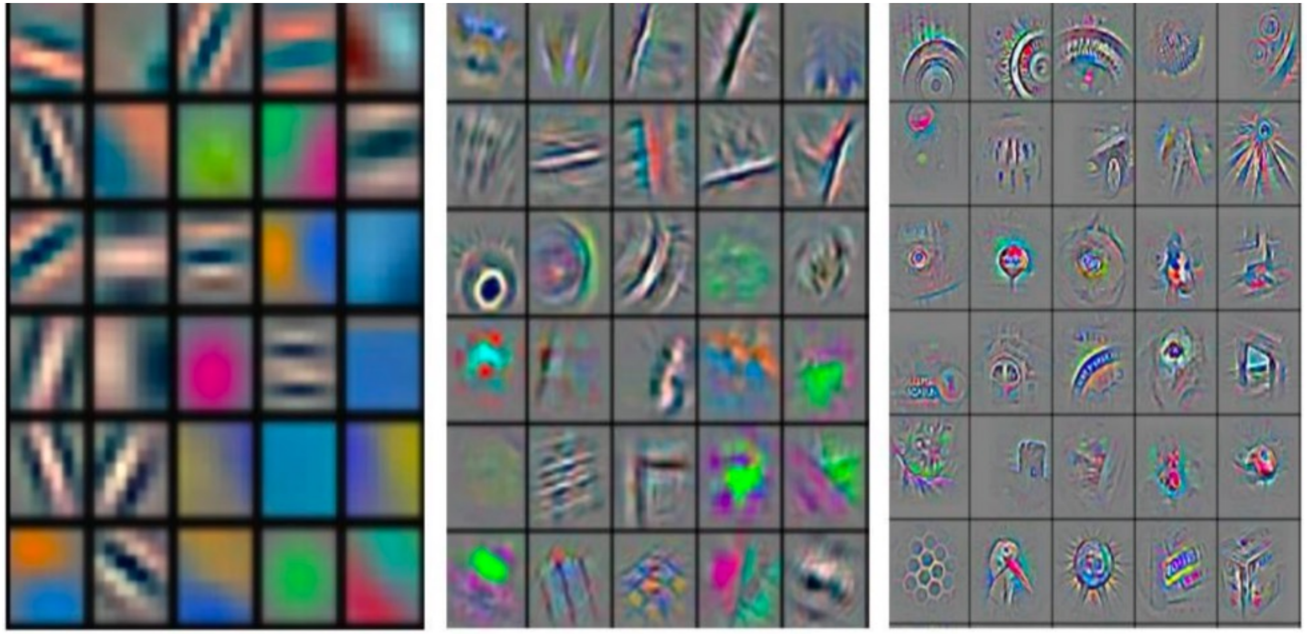


$$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Summary

- Traditional Feature extractor:
  - hand-crafted filters
  - a lot of expertise
  - Limited filters

Can we learn filters automatically?



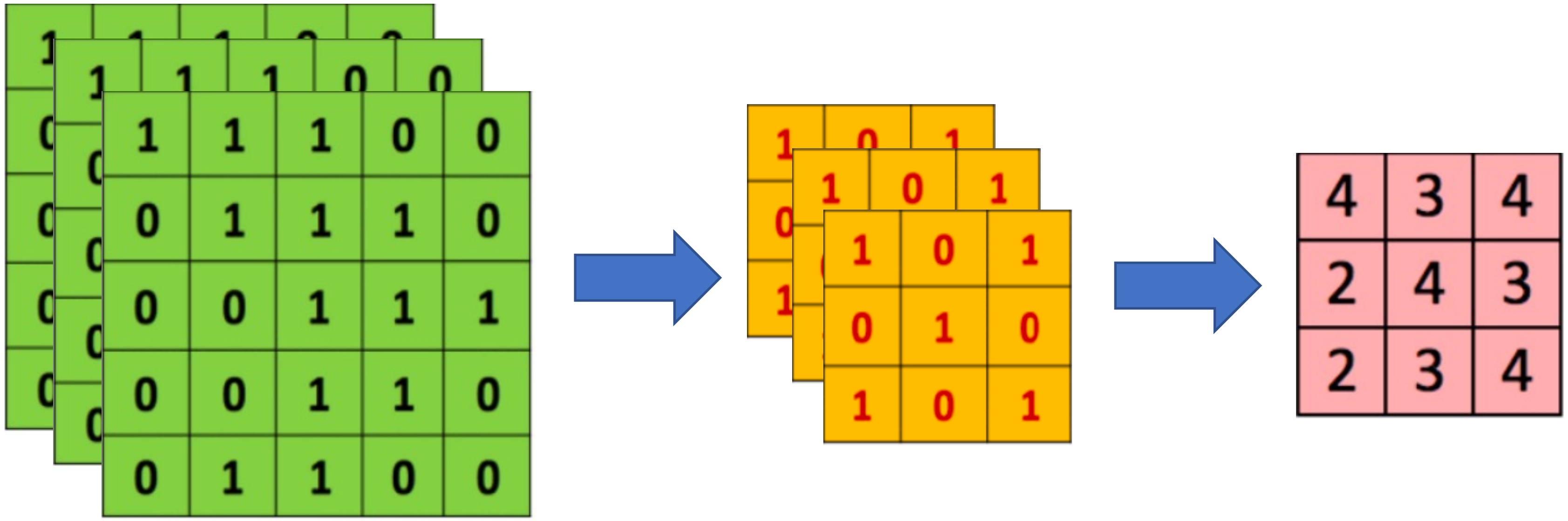
Convolutional Neural Network

- Convolutional layer

$$x_j^\ell = f(\sum_i conv(x_i^{\ell-1}, k_{ij}^\ell) + b_j^\ell)$$

x_j^ℓ is the j-th feature map in the ℓ-th layer,

k_{ij}^ℓ is the convolutional kernel in the ℓ-th layer



3 input feature maps
1 output feature maps
3x1 filters

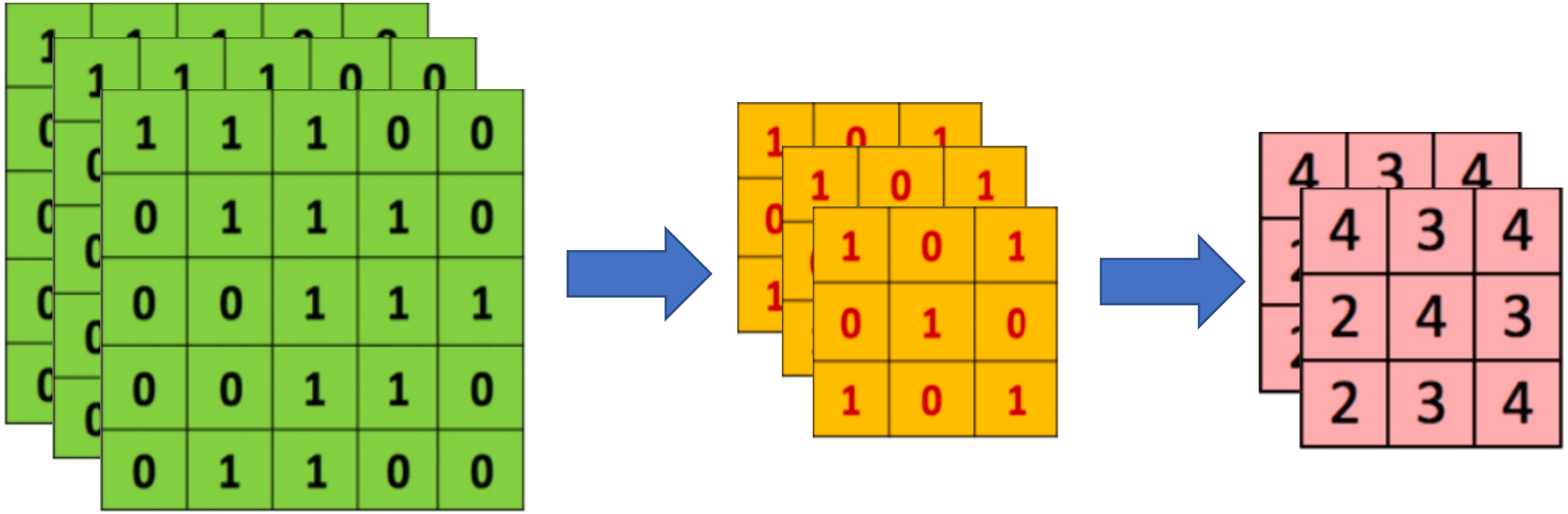
Convolutional Neural Network

- Convolutional layer

$$x_j^\ell = f(\sum_i conv(x_i^{\ell-1}, k_{ij}^\ell) + b_j^\ell)$$

x_j^ℓ is the j-th feature map in the ℓ-th layer,

k_{ij}^ℓ is the convolutional kernel in the ℓ-th layer



3 input feature maps
2 output feature maps
3x2 filters

Convolutional Neural Network

- Question:

Input features: $5 \times 32 \times 32$

Convolution kernel: $8 \times 5 \times 3 \times 3$

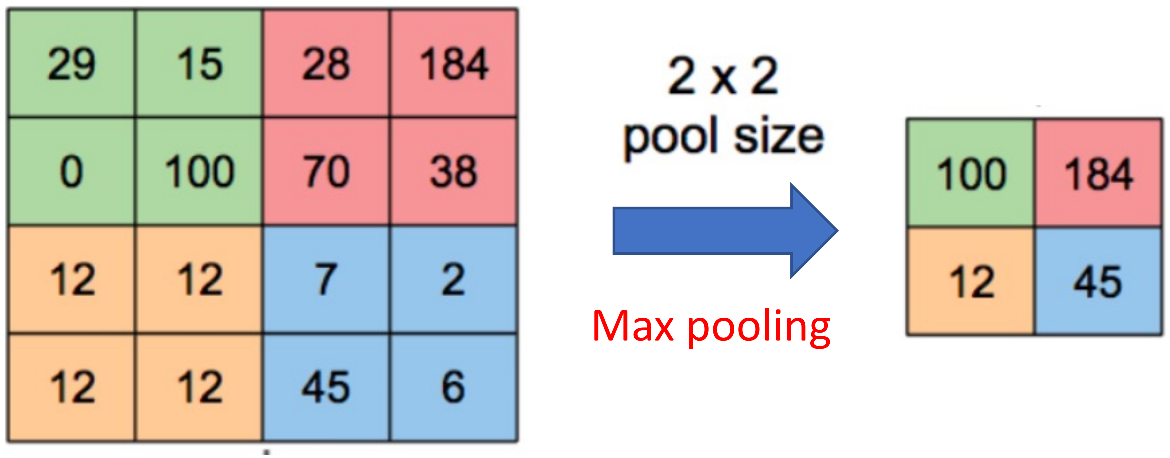
$$x_j^\ell = f(\sum_i conv(x_i^{\ell-1}, k_{ij}^\ell) + b_j^\ell)$$

- How many input channels are there?
- How many output channels are there?

- How many trainable weights are there?

Convolutional Neural Network

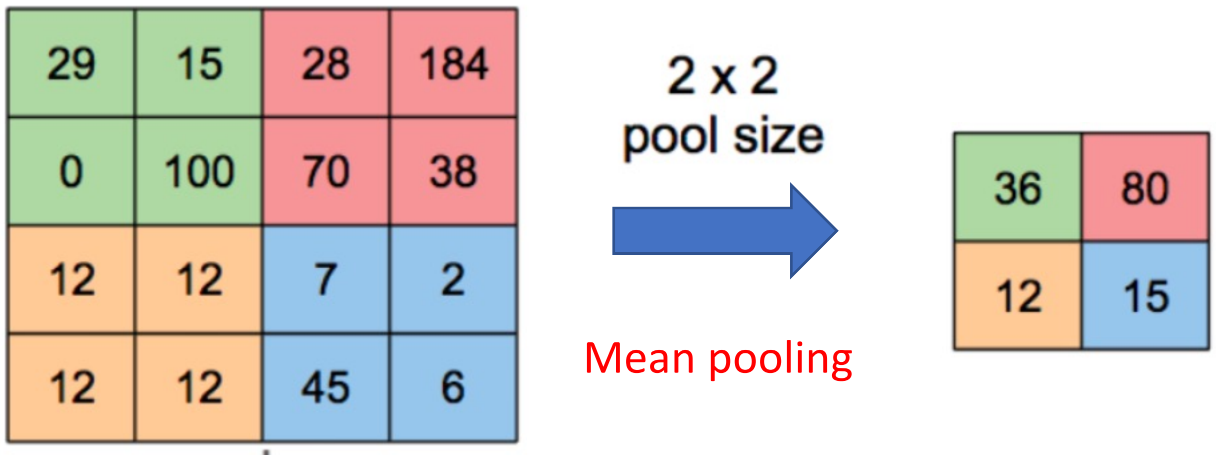
- Pooling layer
  - Reduce the dimensionality of each feature map
  - Max pooling, mean pooling



- Pool size = 2×2.
- No overlap (stride = 2×2).

Convolutional Neural Network

- Pooling layer
  - Reduce the dimensionality of each feature map
  - Max pooling, mean pooling



- Pool size = 2×2.
- No overlap (stride = 2×2).

Summary

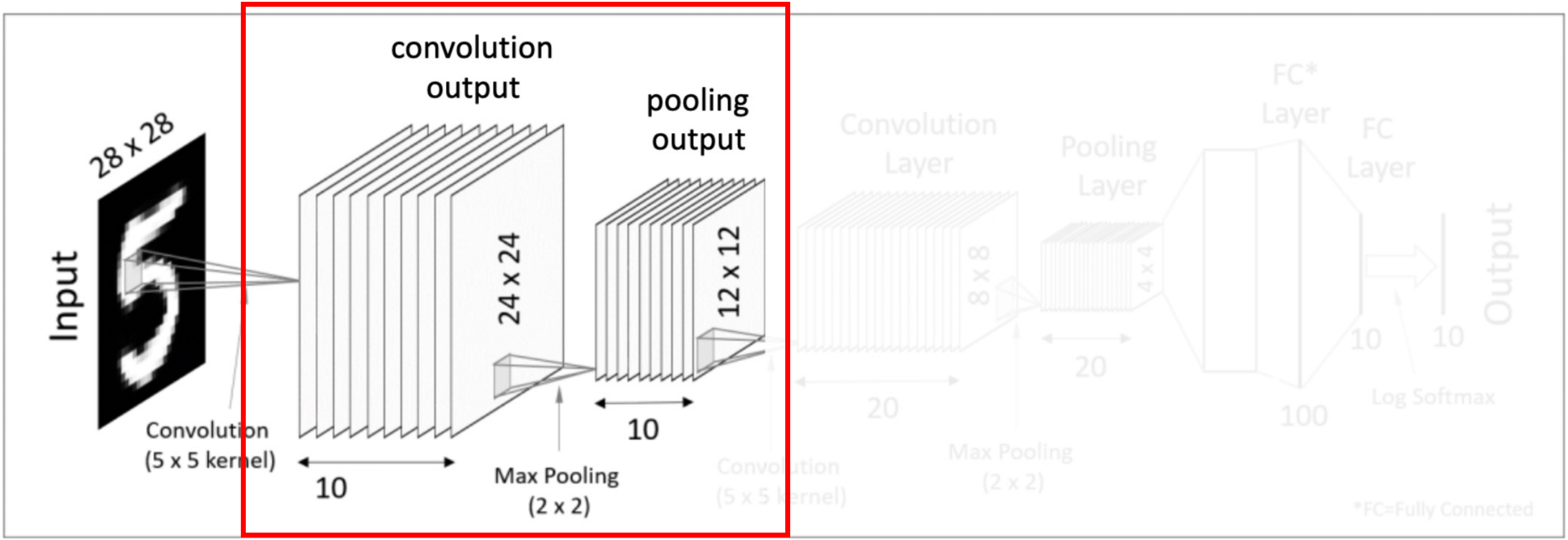
- CNN
  - Convolutional layer

$$x_j^\ell = f(\sum_i conv(x_i^{\ell-1}, k_{ij}^\ell) + b_j^\ell)$$

  - Pooling layer
    - Reduce dimensionality

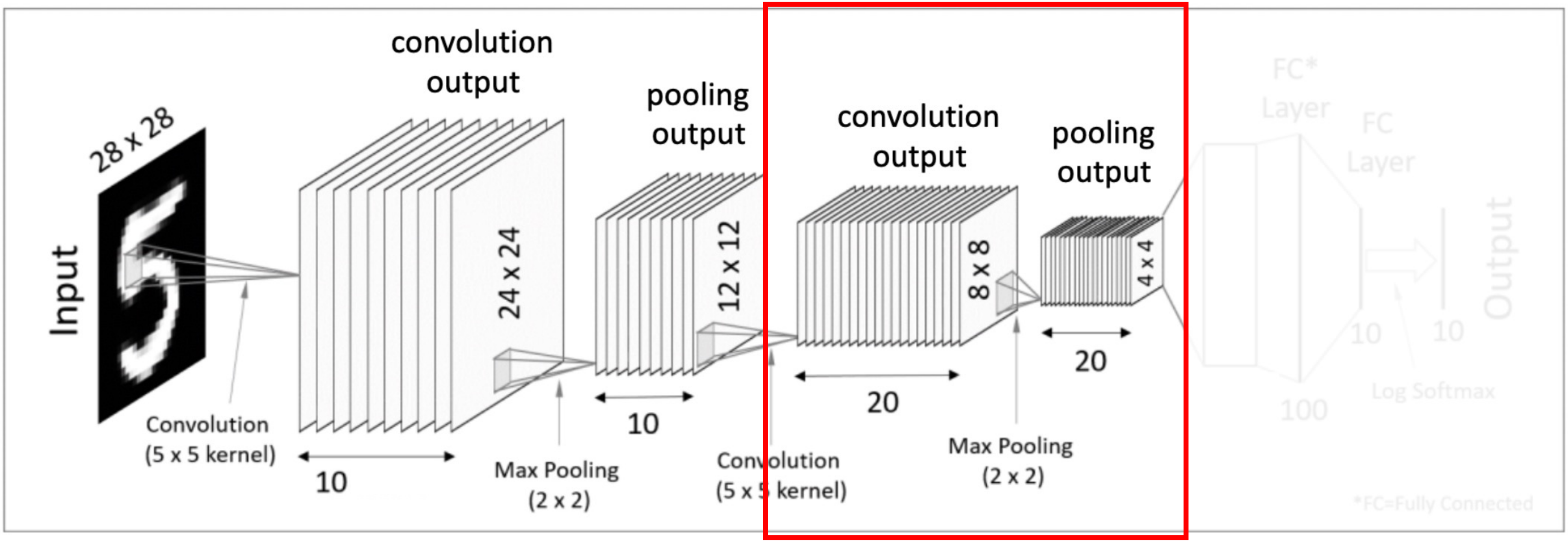
Example

- Convolution + pooling 1



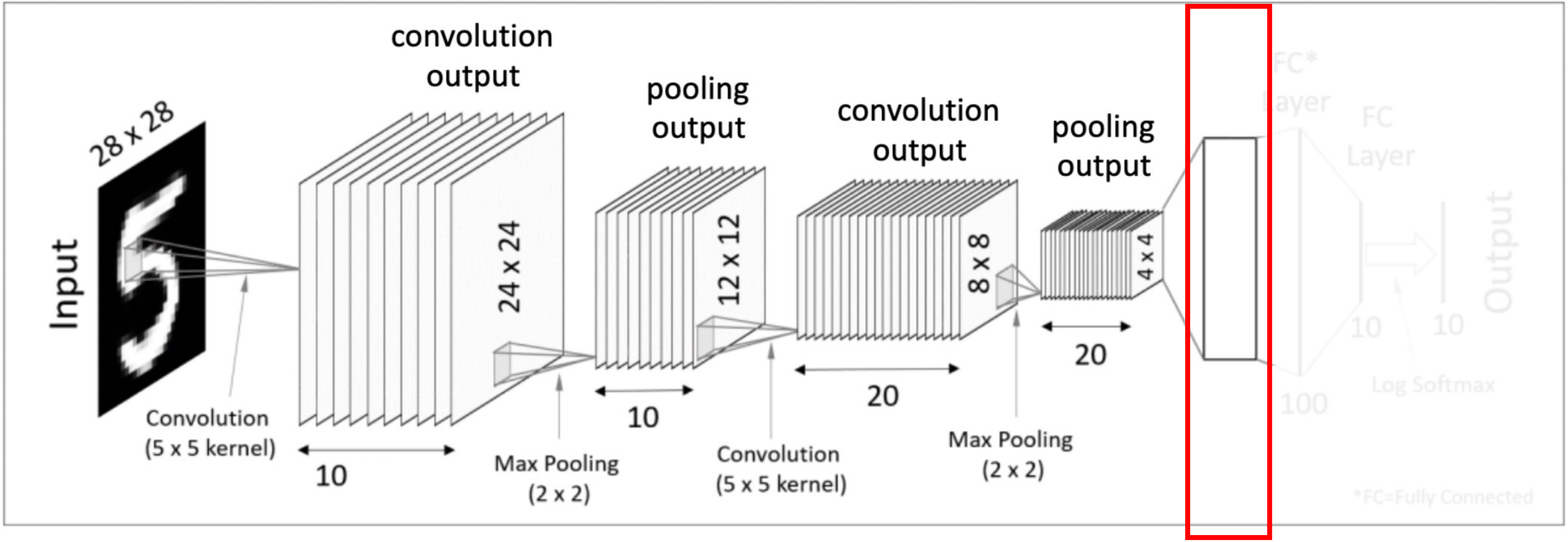
Example

- Convolution + pooling 2



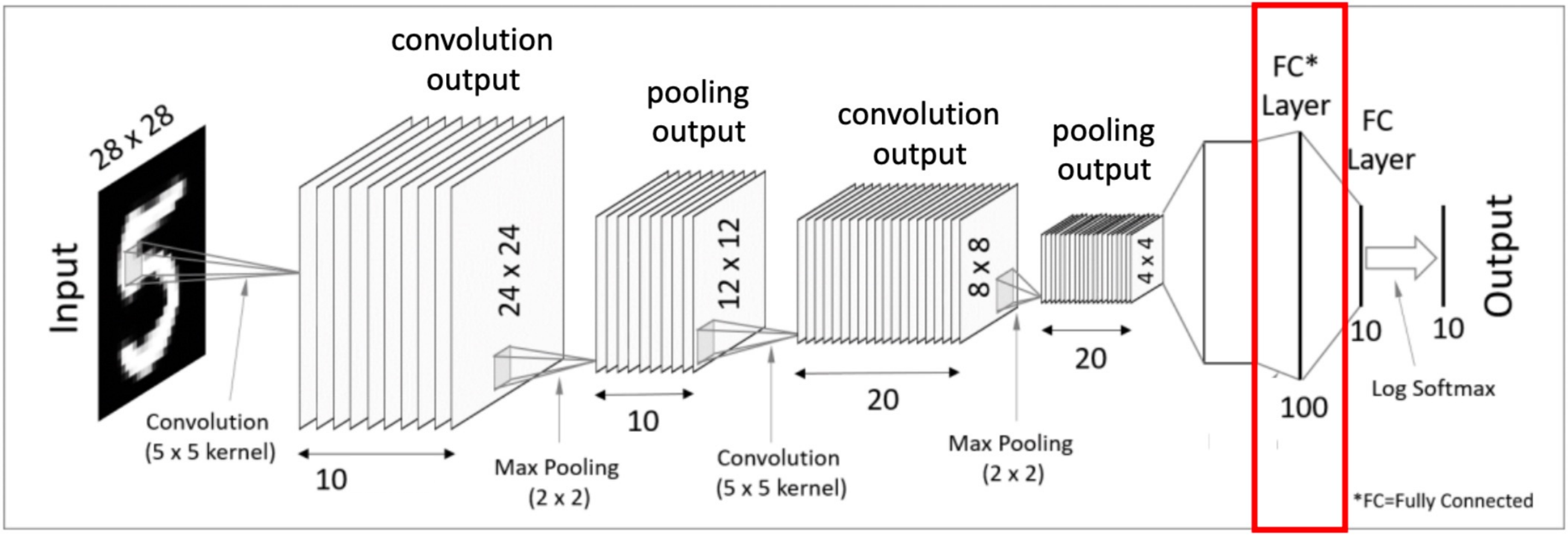
Example

- Flatten the 20 feature maps (4x4)



Example

- Add one fully-connected layer



Question

- How many model parameters?