EI320A(3) 深度學習使用 Python

Instructors

Tipajin Thaipisutikul (t.greentip@gmail.com)

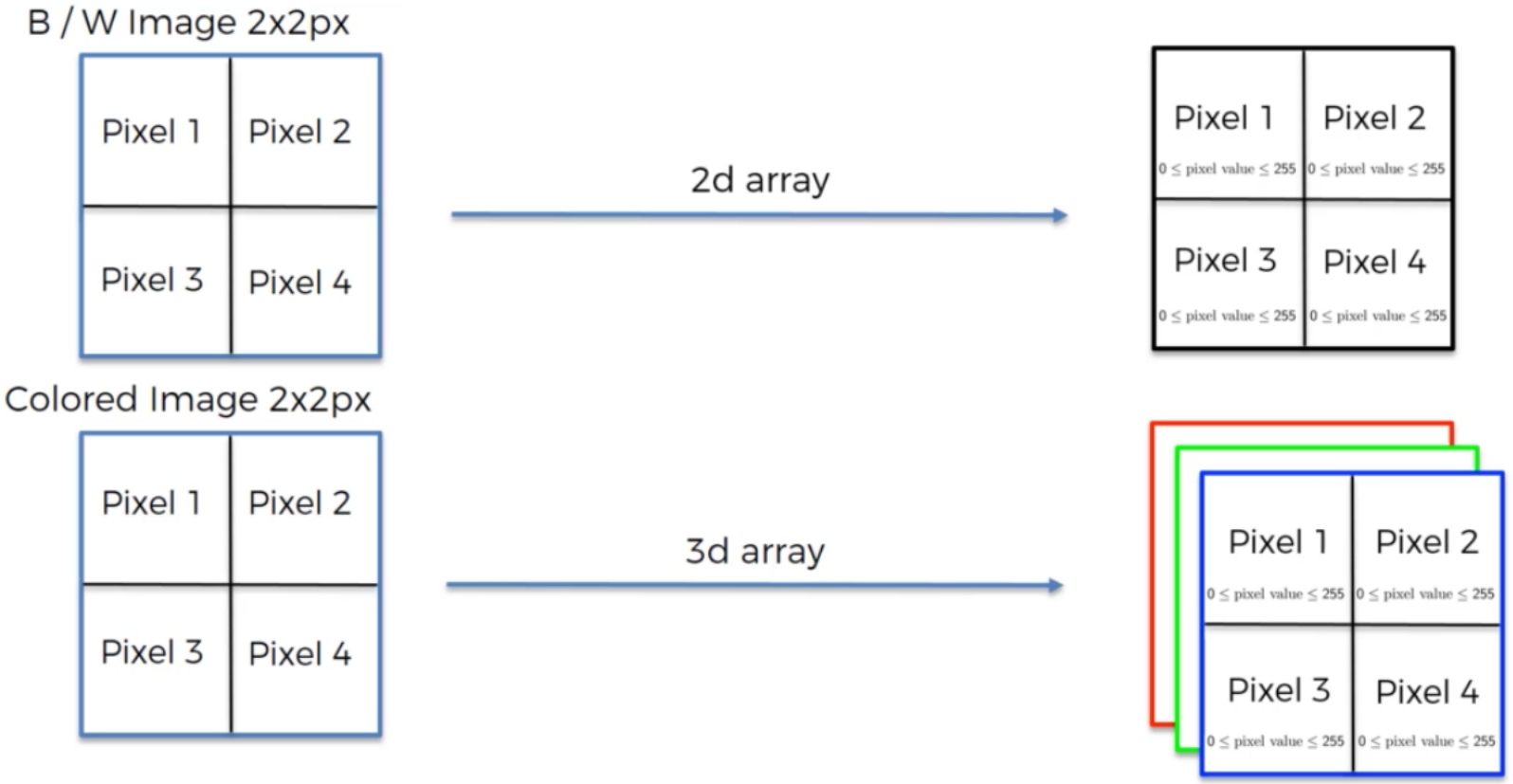
Prof. Huang-Chia Shih (hcshih@Saturn.yzu.edu.tw)

Week	Date	Content	Note	Total
1	2/26	Welcome to the course	Homework (1)	1
2	3/5	Crash Course of Python, NumPy, Pandas, and Matplotlib	In class hands-on (4)	5
3	3/12	Get to know about Data, ML: Classification Models	In class hands-on (5)	10
4	3/19	ML: Regression Models	In class hands-on (5)	15
5	3/26	ML: Clustering/Apriori Models	In class hands-on (5)	20
6	4/2	Holiday		
7	4/9	Introduction to Deep Learning (ANN)		
8	4/16	ANN Labs, Introduction to Convolutional Neural Network (CNN)	In class hands-on (10)	30
9	**4/23**	**Convolutional Neural Network (CNN) & CNN Labs**	**In class hands-on (5)**	**35**
10	4/30	Introduction to Recurrent Neural Network (RNN)	In class hands-on (5)	40
11	5/7	Recurrent Neural Network (RNN) & RNN Labs	In class hands-on (5)	45
12	5/14	Project Proposal Presentation	Proposal Presentation (10)	55
13	5/21	Wrap Up all ANN, CNN, RNN	In class hands-on (5)	60
14	5/28	NLP & Attention Neural Network	In class hands-on (5)	65
15	6/4	Generative Adversarial Network (GAN)	In class hands-on (5)	70
16	6/11	Reinforcement Learning (RL)	In class hands-on (5)	75
17	6/18	Final Project Presentation	Final Presentation (30)	105

CNN Outline

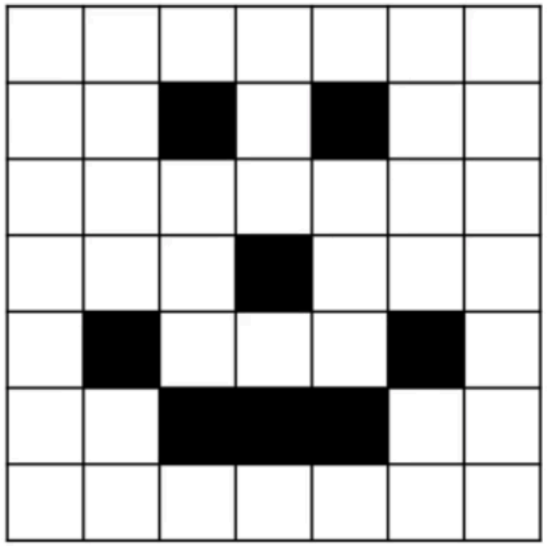
- What are Convolutional Neural Networks?
- Step 1 - Convolutional Operation
- Step 1(b) – ReLu Layer
- Step 2 – Pooling
- Step 3 - Flattening
- Step 4 – Full Connection
- Summary

Convolutional Neural Networks



0: black
255:white

Convolutional Neural Networks



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

STEP 1: Convolution

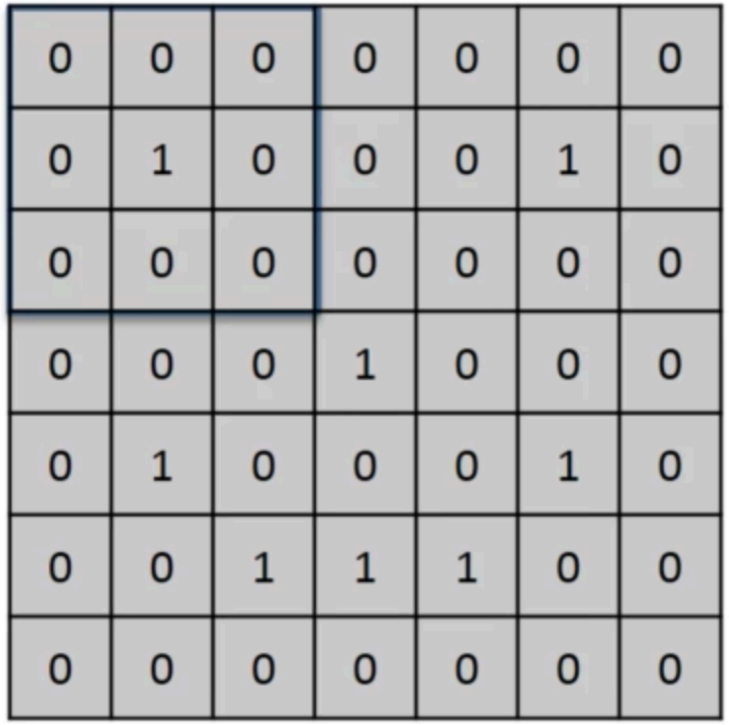
STEP 2: Max Pooling

STEP 3: Flattening

STEP 4: Full Connection

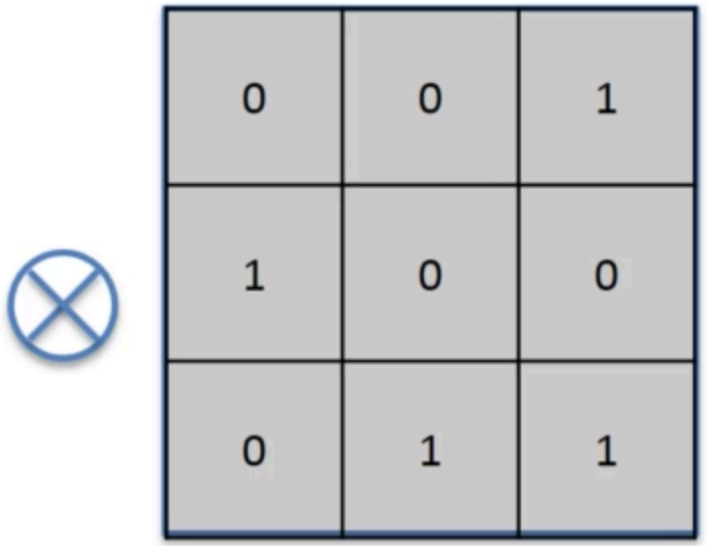
Step 1 - Convolutional Operation

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$



Input Image

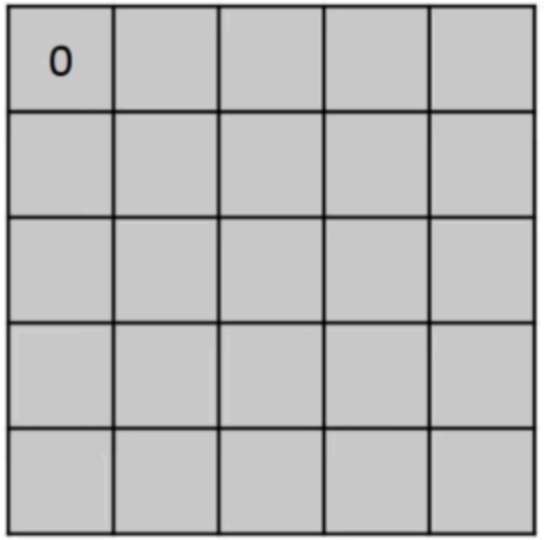
Element-Wise



Feature Detector

Filter/ Kernel

Stride=1



Feature Map

Activation Map

Is smaller than an original image

Step 1 - Convolutional Operation

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$

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

Input Image

Element-Wise ⊗

0	0	1
1	0	0
0	1	1

Feature Detector

Filter/ Kernel

Stride=1

=

0	1	0	0	0
0				

Feature Map

Activation Map

Is smaller than an original image

Step 1 - Convolutional Operation

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$

Stride=1

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

Input Image

Element-Wise

0	0	1
1	0	0
0	1	1

Feature Detector

Filter/ Kernel

=

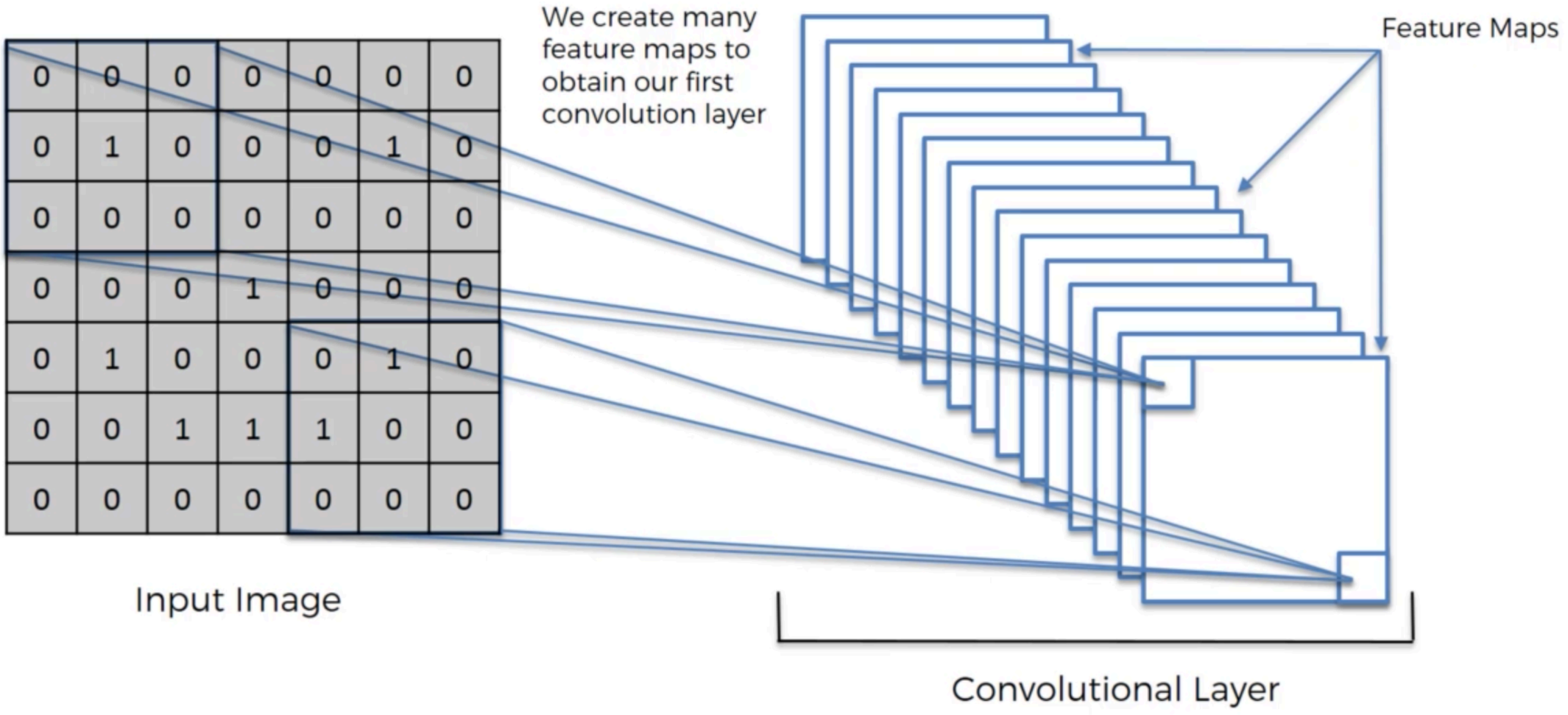
0	1	0	0	0
0	1	1	1	0
1	0	1	2	1
1	4	2	1	0
0	0	1	2	1

Feature Map

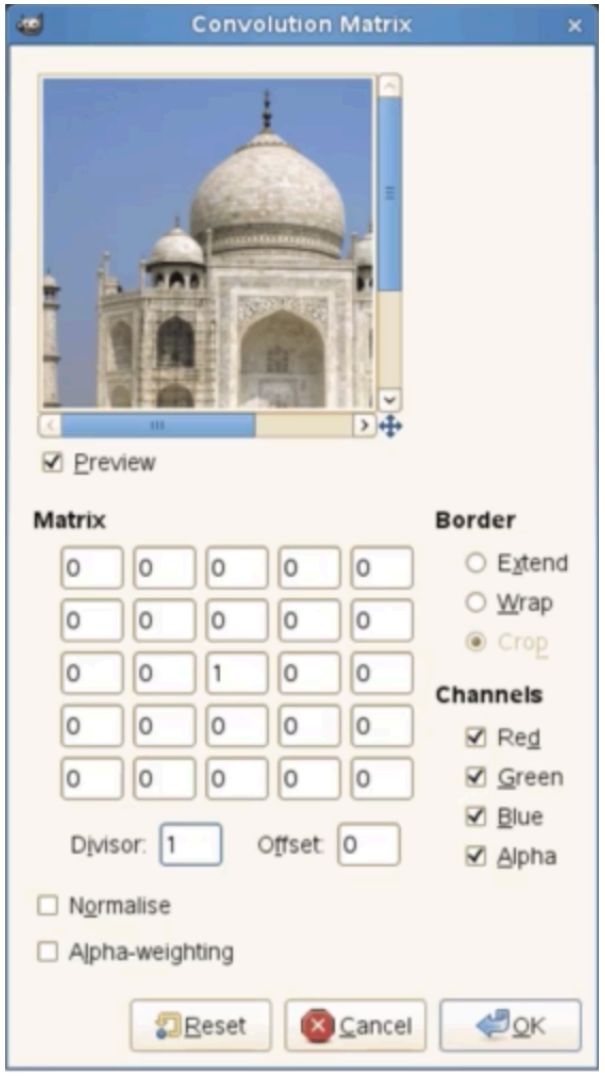
Activation Map

Is smaller than an original image

Step 1 - Convolutional Operation



Step 1 - Convolutional Operation



Blur:

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

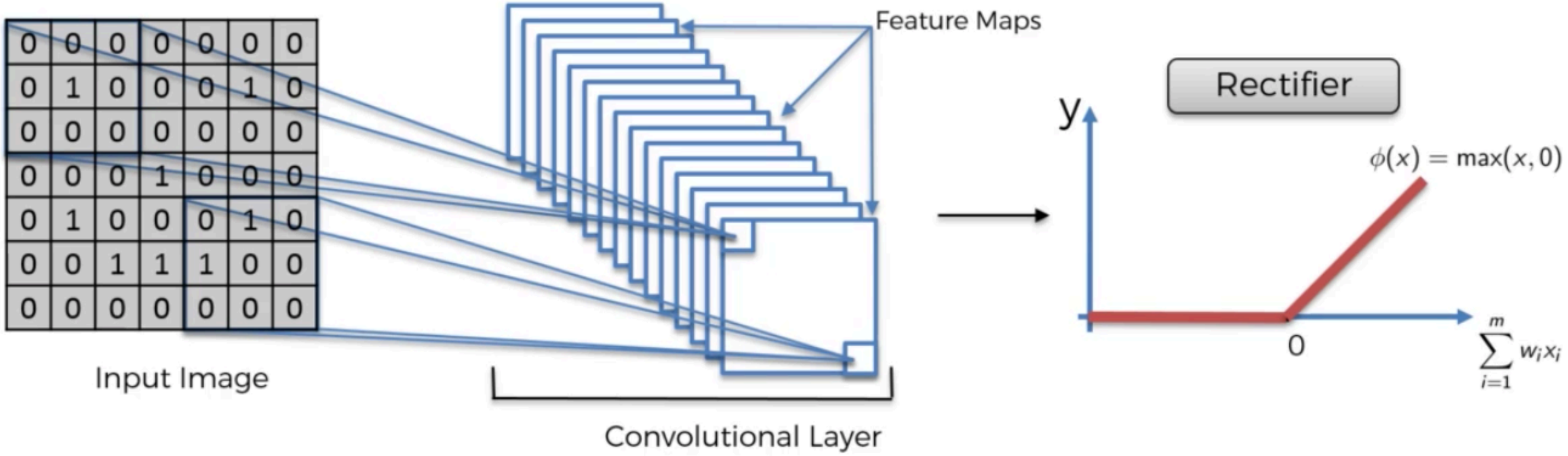
Edge Detect:

	0	1	0	
	1	-4	1	
	0	1	0	

CNN Outline

- What are Convolutional Neural Networks?
- Step 1 - Convolutional Operation
- Step 1(b) – ReLu Layer
- Step 2 – Pooling
- Step 3 - Flattening
- Step 4 – Full Connection
- Summary

Step 1(b) – ReLu Layer



Step 1(b) – ReLu Layer

Without ReLu

With ReLu

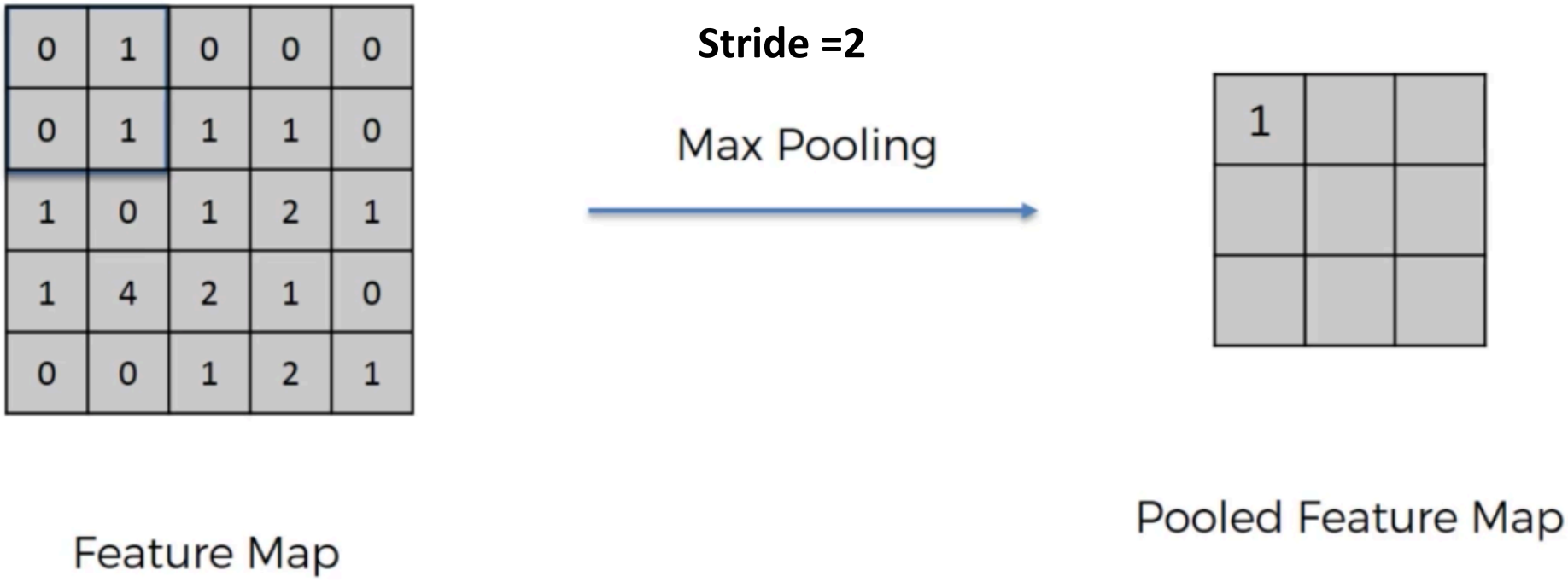
CNN Outline

- What are Convolutional Neural Networks?
- Step 1 - Convolutional Operation
- Step 1(b) – ReLu Layer
- Step 2 – Pooling
- Step 3 - Flattening
- Step 4 – Full Connection
- Summary

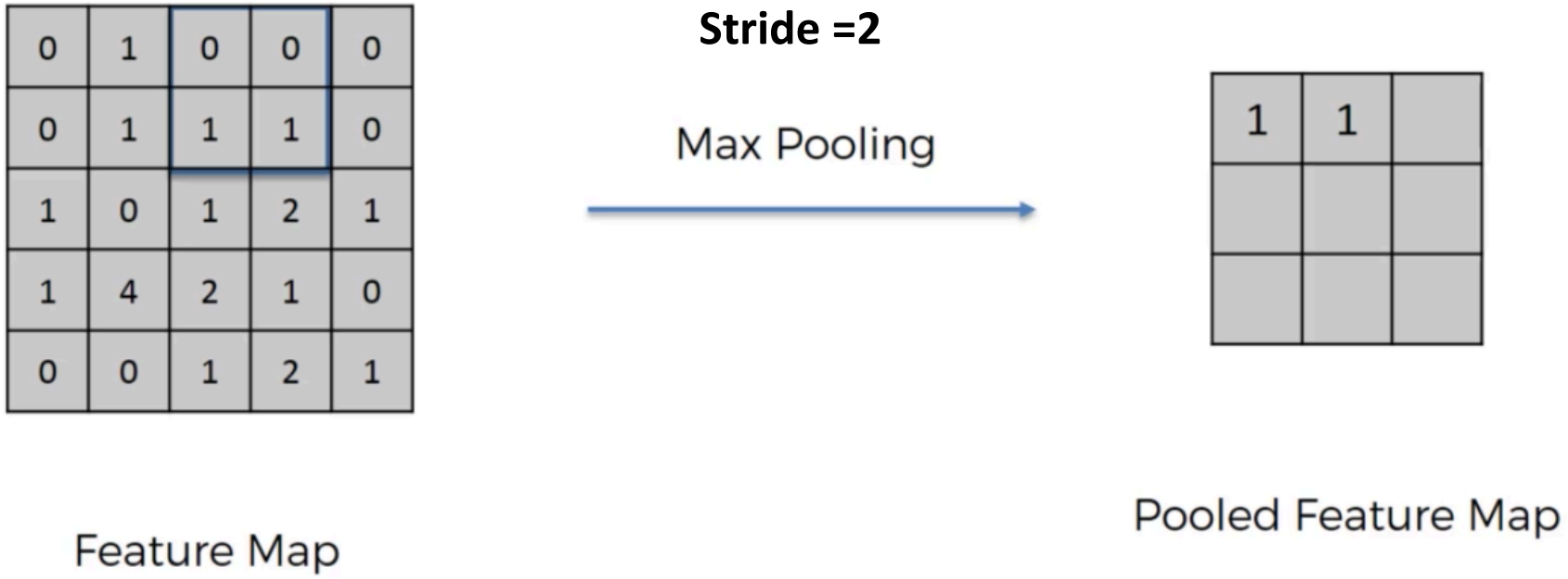
Step 2 – Pooling

Image Source: Wikipedia

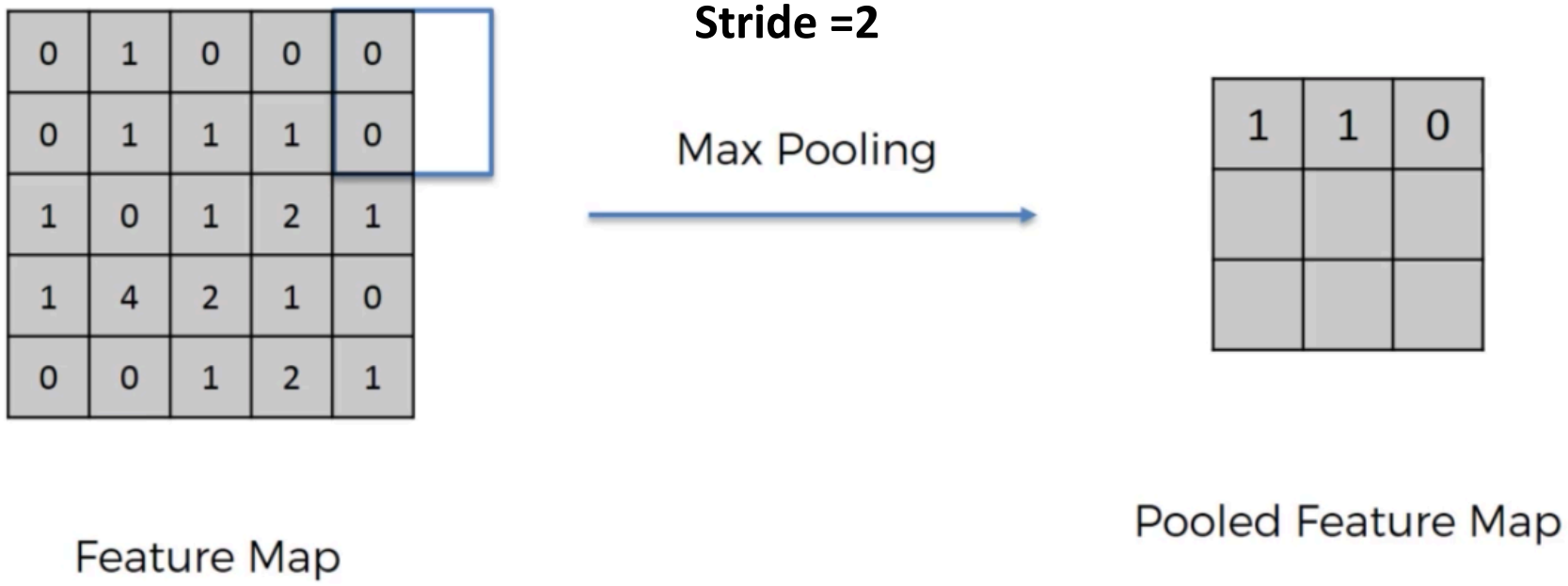
Step 2 – Max Pooling



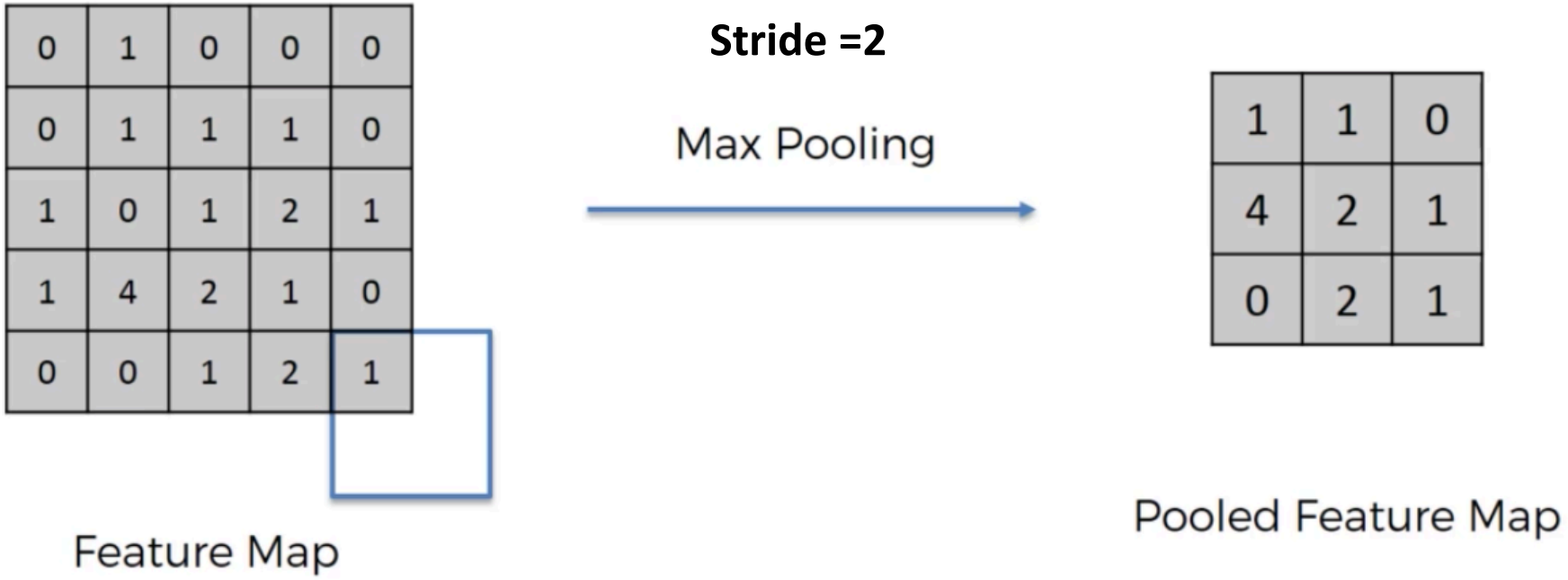
Step 2 – Max Pooling



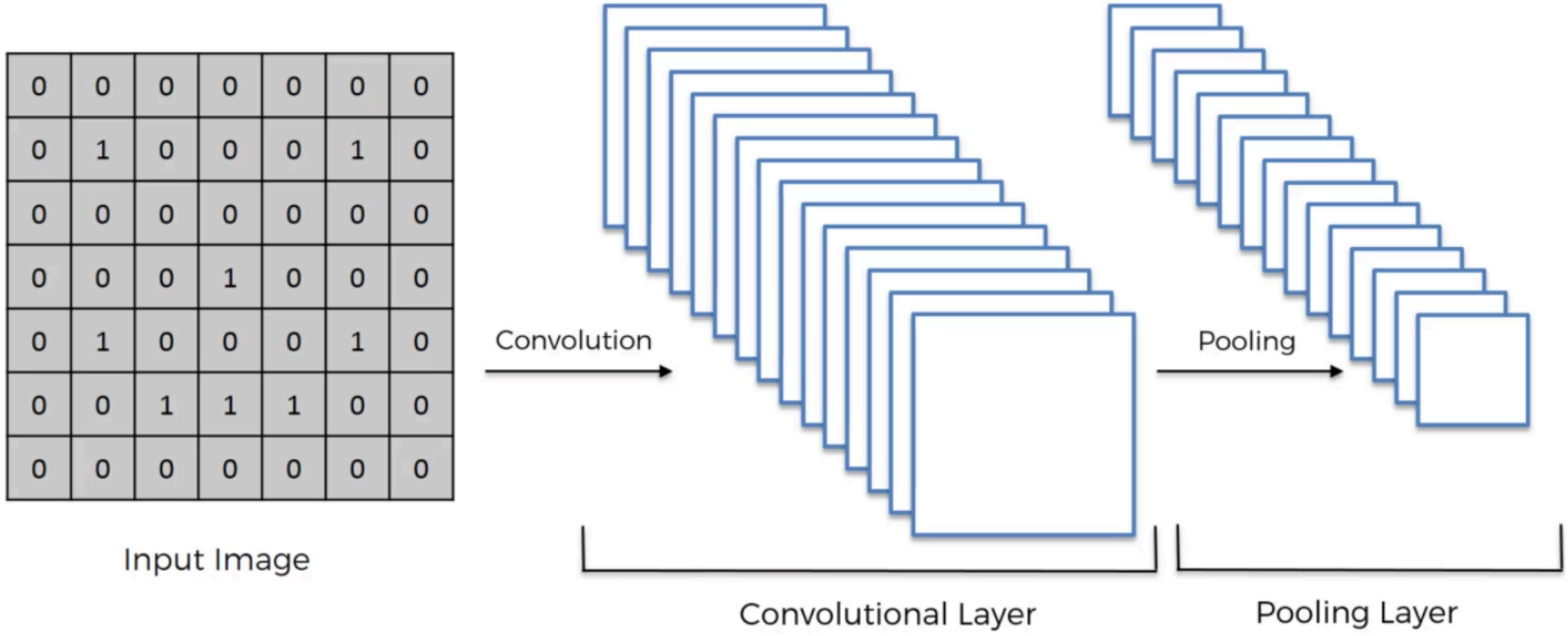
Step 2 – Max Pooling



Step 2 – Max Pooling



Step 2 – Max Pooling



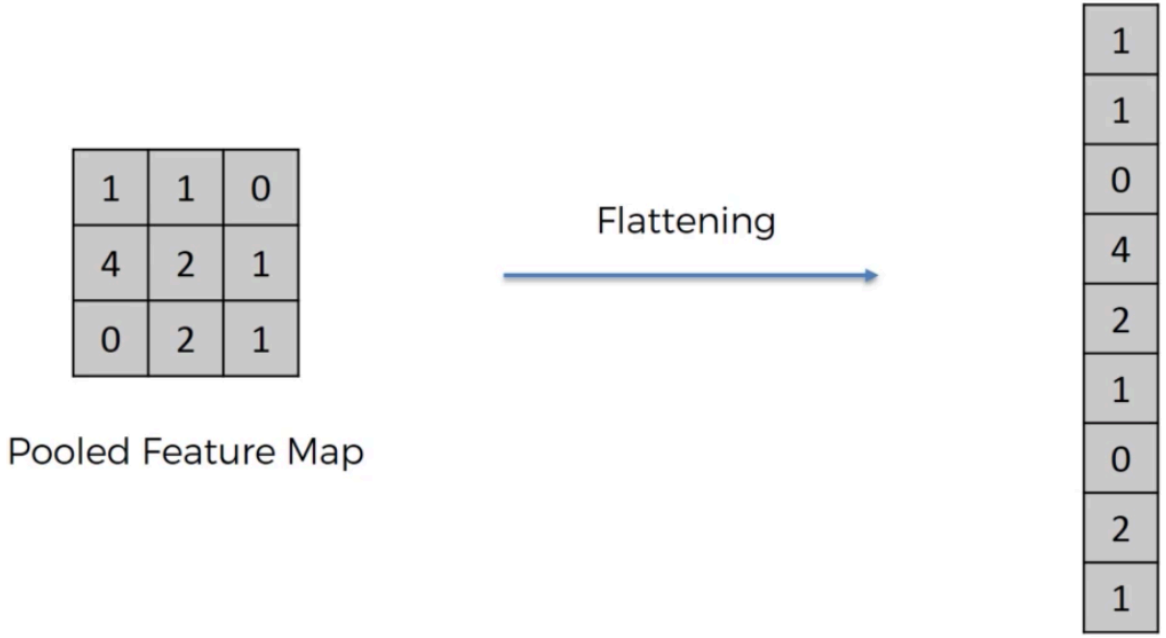
Step 2 – Max Pooling

https://www.cs.ryerson.ca/~aharley/vis/conv/flat.html

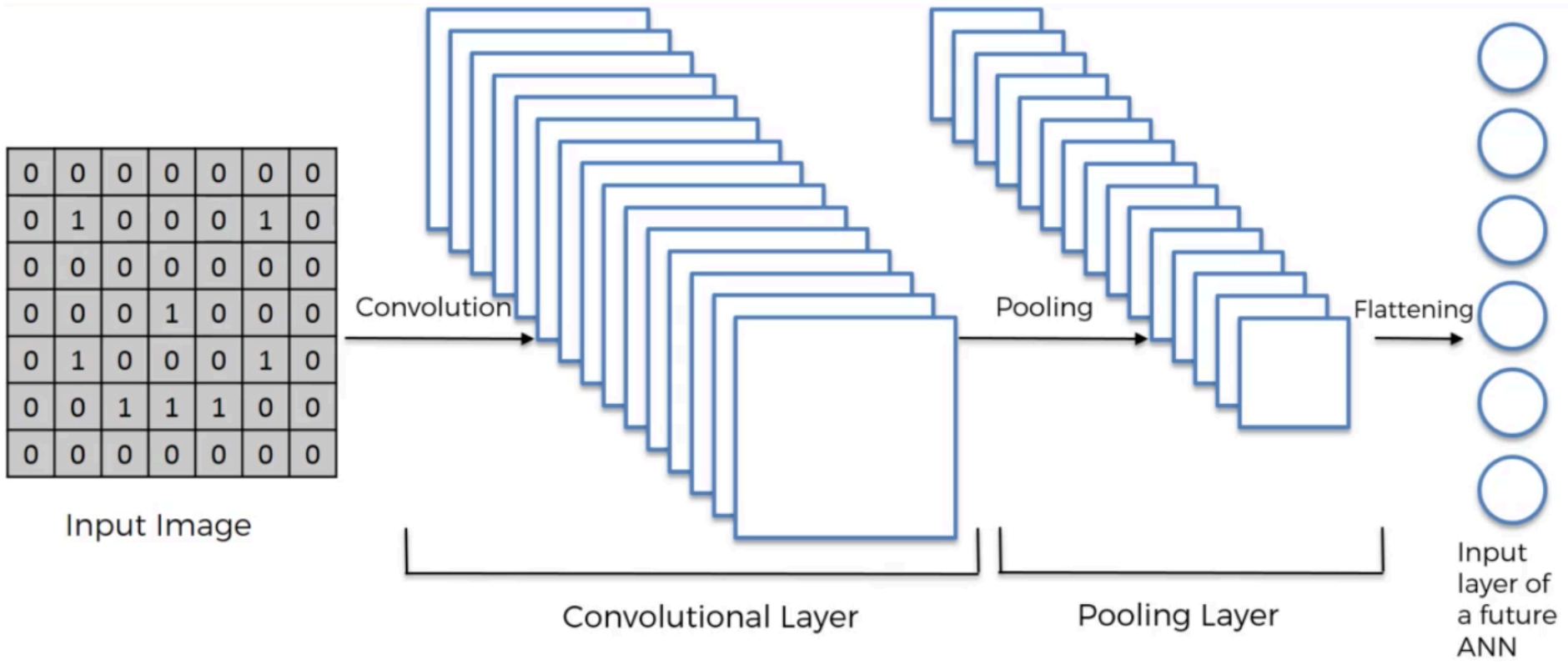
CNN Outline

- What are Convolutional Neural Networks?
- Step 1 - Convolutional Operation
- Step 1(b) – ReLu Layer
- Step 2 – Pooling
- Step 3 - Flattening
- Step 4 – Full Connection
- Summary

Step 3 - Flattening



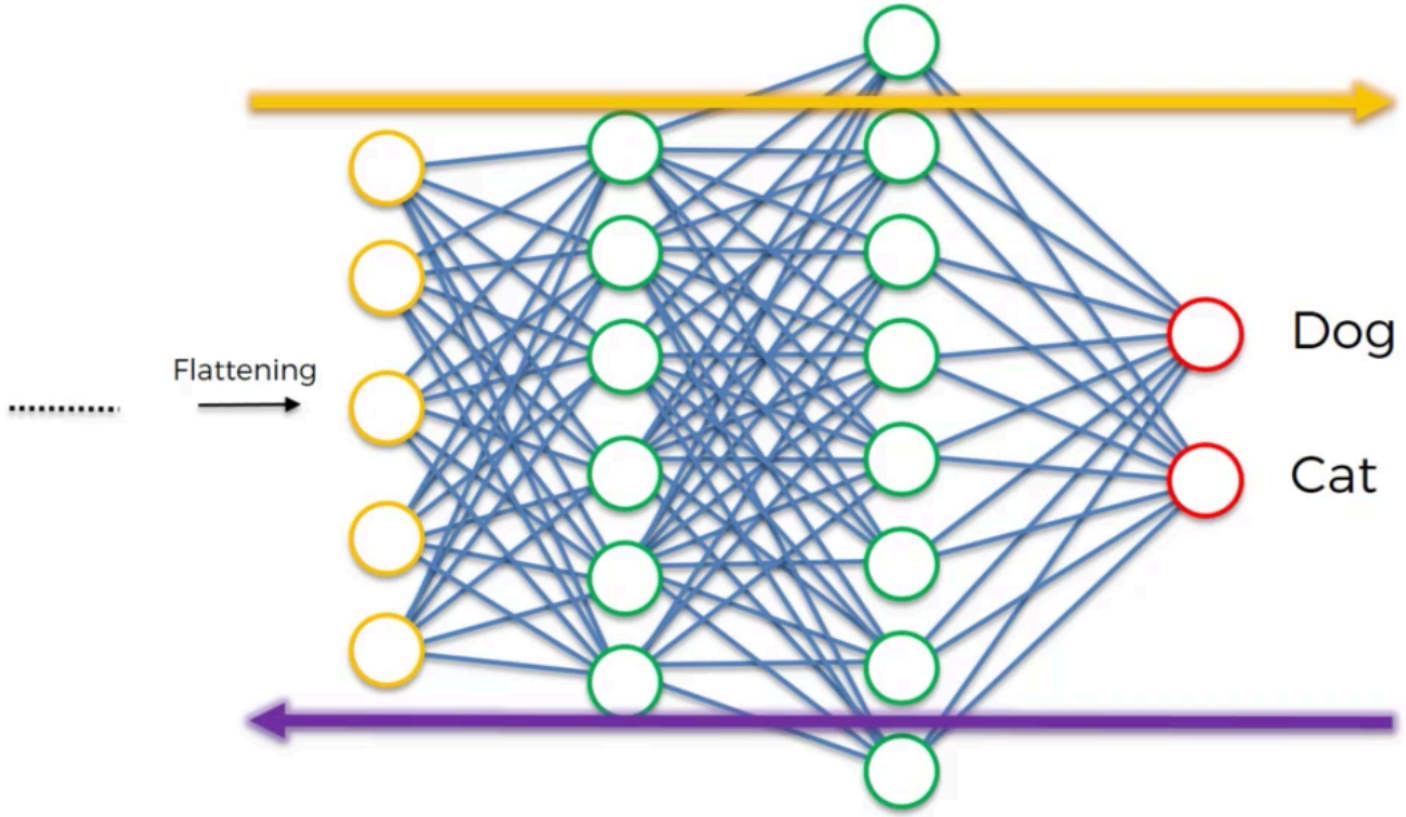
Step 3 - Flattening



CNN Outline

- What are Convolutional Neural Networks?
- Step 1 - Convolutional Operation
- Step 1(b) – ReLu Layer
- Step 2 – Pooling
- Step 3 - Flattening
- Step 4 – Full Connection
- Summary

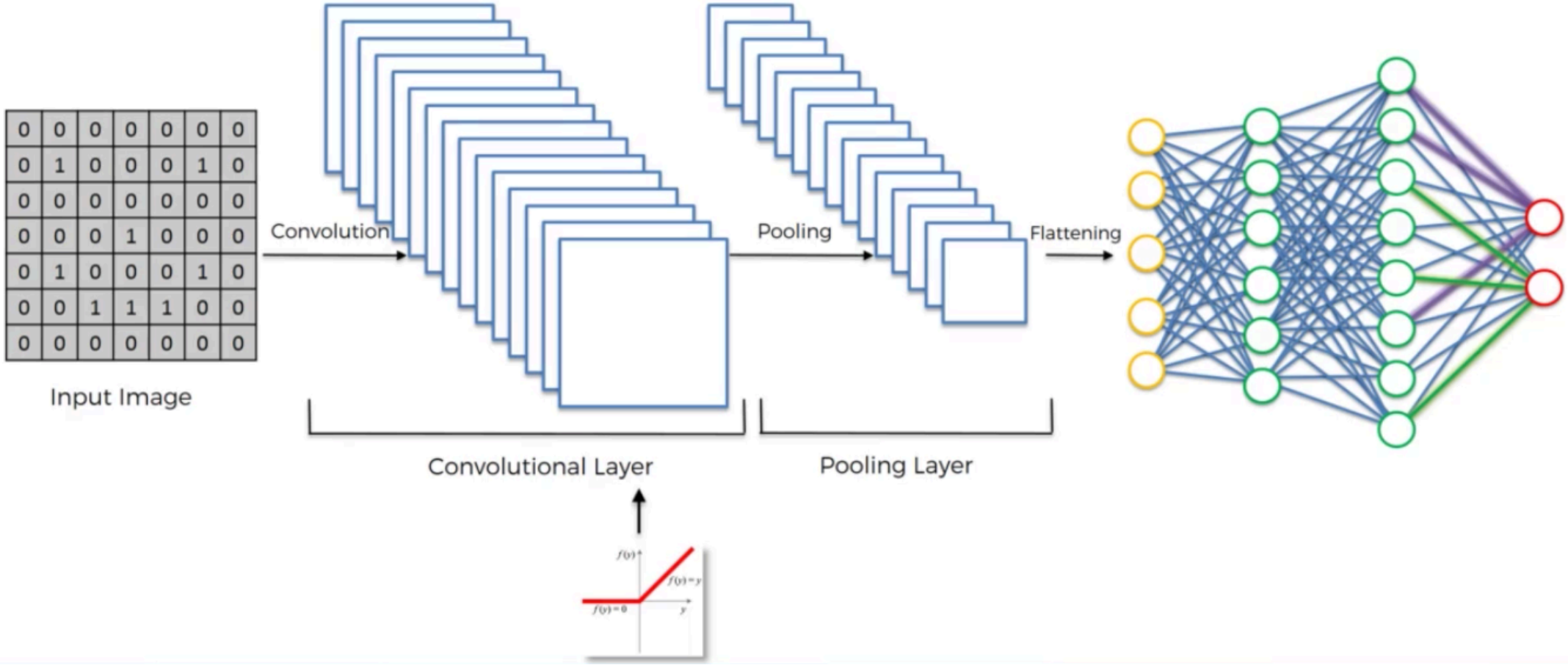
Step 4 – Full Connection



CNN Outline

- What are Convolutional Neural Networks?
- Step 1 - Convolutional Operation
- Step 1(b) – ReLu Layer
- Step 2 – Pooling
- Step 3 - Flattening
- Step 4 – Full Connection
- Summary

Summary


0 0 0 0 0 0 0
0 1 0 0 0 1 0
0 0 0 0 0 0 0
0 0 0 1 0 0 0
0 1 0 0 0 1 0
0 0 1 1 1 0 0
0 0 0 0 0 0 0
Input Image
Convolution
Pooling
Flattening
Convolutional Layer
Pooling Layer

CNN

- Convolutional Neural Networks are a powerful artificial neural network technique. These networks preserve the spatial structure of the problem and were developed for object recognition tasks such as handwritten digit recognition.
- The benefits of using convolutional neural networks:
  - They use fewer parameters (weights) to learn than a fully connected network.
  - They are designed to be invariant to object position and distortion in the scene.
  - They automatically learn and generalize features from the input domain.
- There are three types of layers in a Convolutional Neural Network:
  1. Convolutional Layers.
  2. Pooling Layers.
  3. Fully-Connected Layers.

CNN

1. Convolutional layers are comprised of filters and feature maps.

1.1. Filters

- The input size is a fixed square called a patch or a receptive field. If the convolutional layer is an input layer, then the input patch will be pixel values. If they deeper in the network architecture, then the convolutional layer will take input from a feature map from the previous layer.

1.2. Feature Maps

- The feature map is the output of one filter applied to the previous layer. A given filter is drawn across the entire previous layer and moved one pixel at a time.
- Each position results in an activation of the neuron and the output is collected in the feature map
- The distance that filter is moved across the input from the previous layer each activation is referred to as the **stride**. If the size of the previous layer is not cleanly divisible by the size of the filter's receptive field and the size of the stride then it is possible for the receptive field to attempt to read 0 the edge of the input feature map.
- In this case, techniques like **zero padding** can be used to invent mock inputs with zero values for the receptive field to read.

CNN

2. Pooling Layers

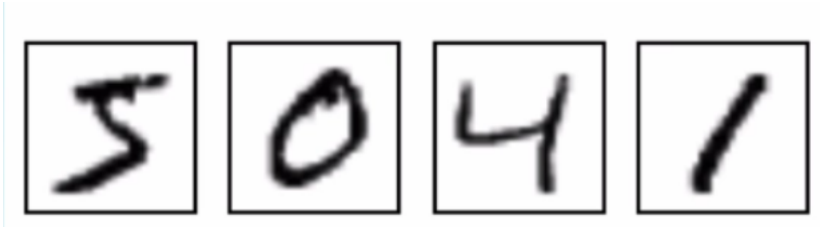
- The pooling layers down-sample the previous layers feature map.
- Pooling layers follow a sequence of one or more convolutional layers and are intended to consolidate the features learned and expressed in the previous layer's feature map.
- As such, **pooling may be considered a technique to compress or generalize feature representations and generally reduce the overfitting of the training data by the model.**
- They too have a receptive field, often much smaller than the convolutional layer.
- Also, the stride or number of inputs that the receptive field is moved for each activation is often equal to the size of the receptive field to avoid any overlap.
- Pooling layers are often very simple, taking the average or the maximum of the input value in order to create its own feature map.

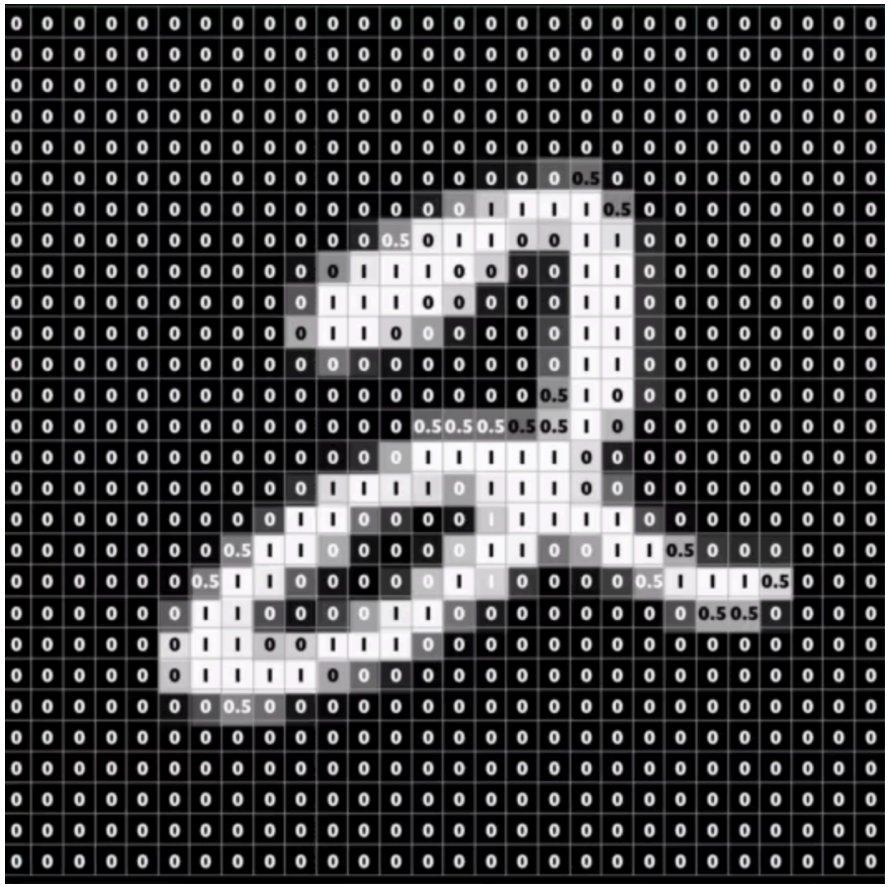
CNN

3. Fully Connected Layers

- Fully connected layers are the normal flat feedforward neural network layer. These layers may have a nonlinear activation function or **a softmax activation in order to output probabilities of class predictions.**
- Fully connected layers are used at the end of the network after feature extraction and consolidation has been performed by the convolutional and pooling layers.
- They are used to create final nonlinear combinations of features and for making predictions by the network.

Demo: MNIST





28 x 28 image => 784 input pixels array