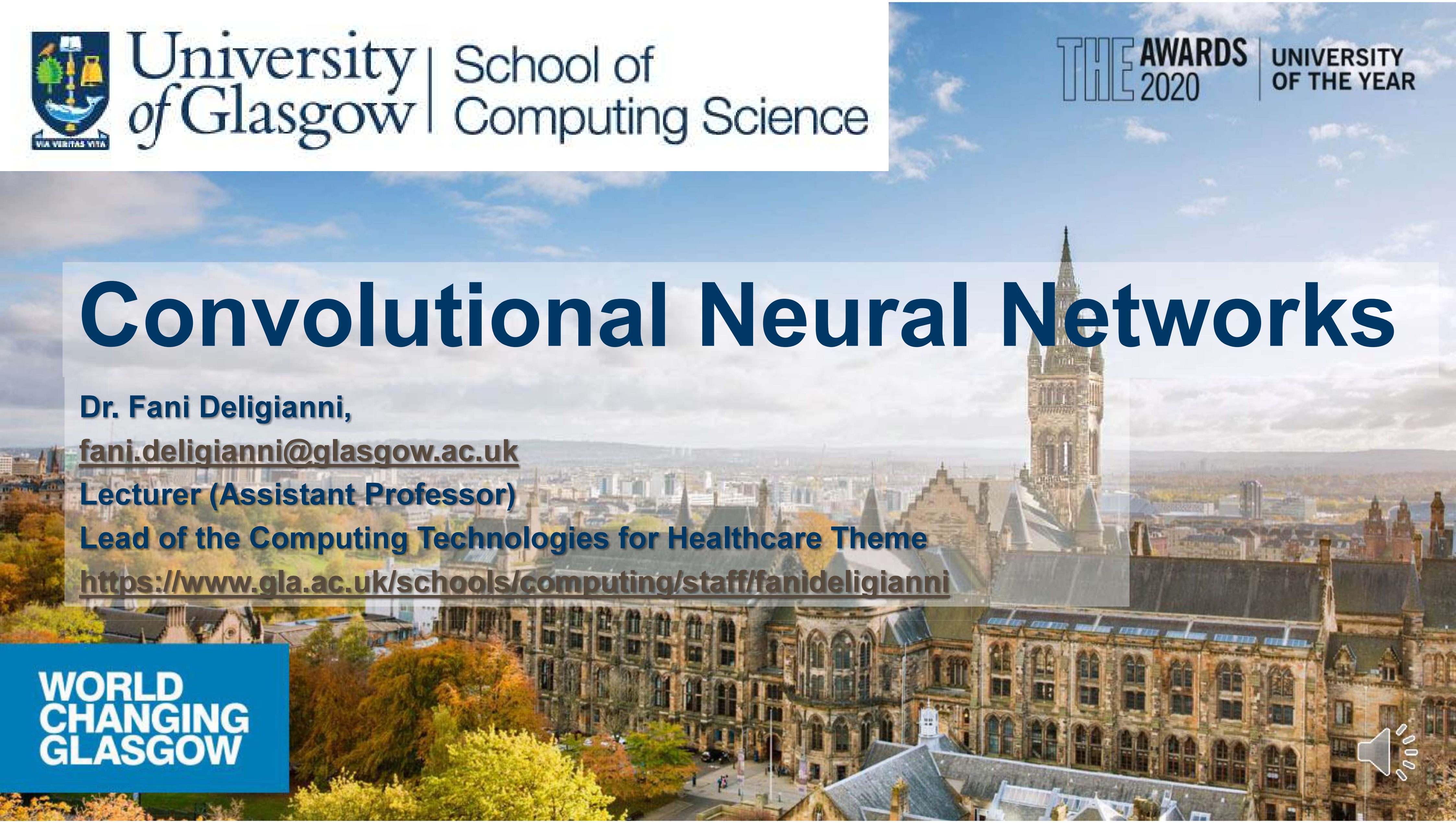
University of Glasgow | School of Computing Science

THE AWARDS 2020 | UNIVERSITY OF THE YEAR



Convolutional Neural Networks

Dr. Fani Deligianni,

fani.deligianni@glasgow.ac.uk

Lecturer (Assistant Professor)

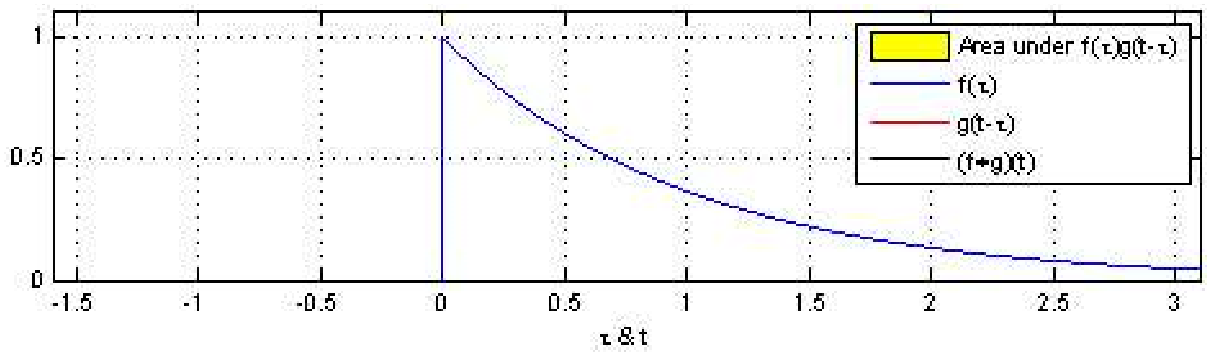
Lead of the Computing Technologies for Healthcare Theme

https://www.gla.ac.uk/schools/computing/staff/fanideligianni

WORLD CHANGING GLASGOW

Convolution

- Convolution is a mathematical operation



Continuous Domain:

$$y(t) = (f * g)(t) = \int_{-\infty}^{\infty} f(\tau) g(t-\tau) d\tau$$

Discrete Domain:

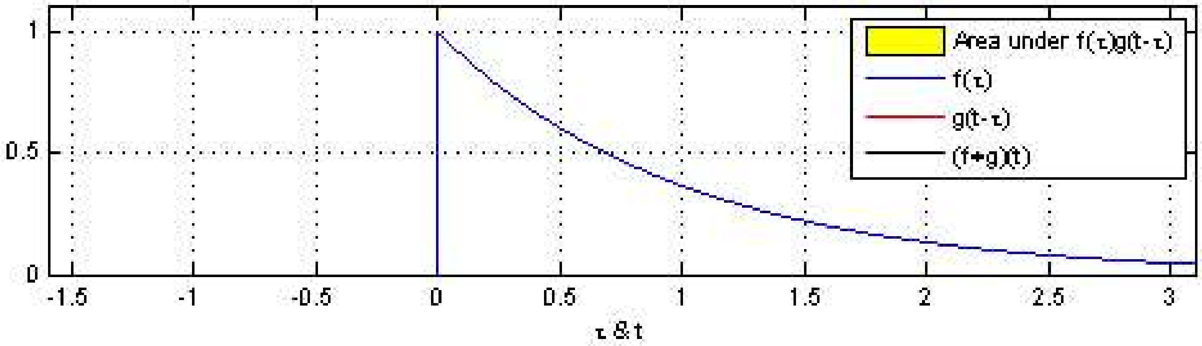
$$y(t) = (f * g)(t) = \sum f(\tau) g(t-\tau)$$

Input function **Kerner/Filter**

https://en.wikipedia.org/wiki/Convolution

Convolution vs Cross-Correlation

- Convolution is a mathematical operation



https://en.wikipedia.org/wiki/Convolution

Continuous Domain:

$$y(t) = (f \ * g)(t) = \int_{\infty}^{\infty} f(\tau)g(t-\tau)d\tau$$

Discrete Domain:

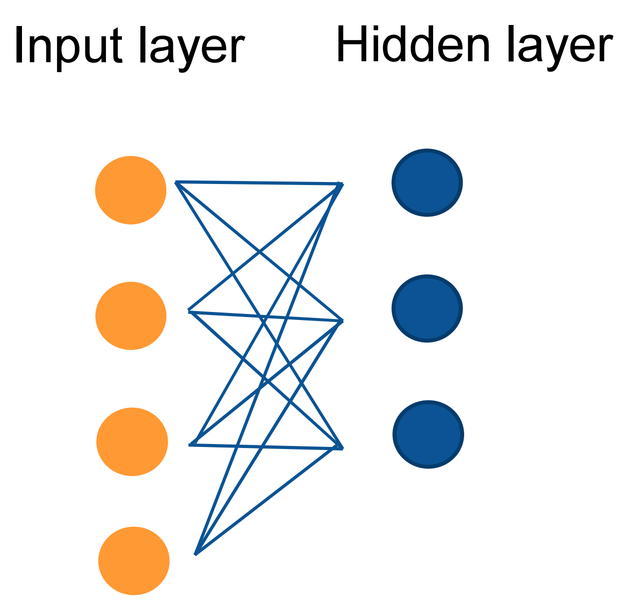
$$y(t) = (f \ * g)(t) = \sum f(\tau)g(t-\tau)$$

Kerner/Filter **Input function**

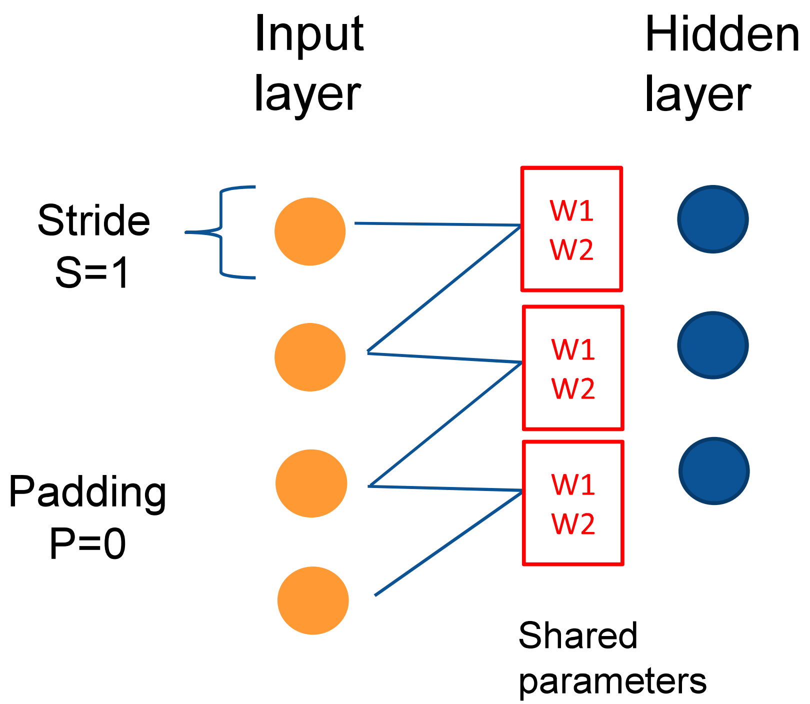
Cross-Correlation:

$$y(t) = (f \otimes g)(t) = \sum f(\tau)g(t+\tau)$$

Convolutional Neural Networks vs MLP

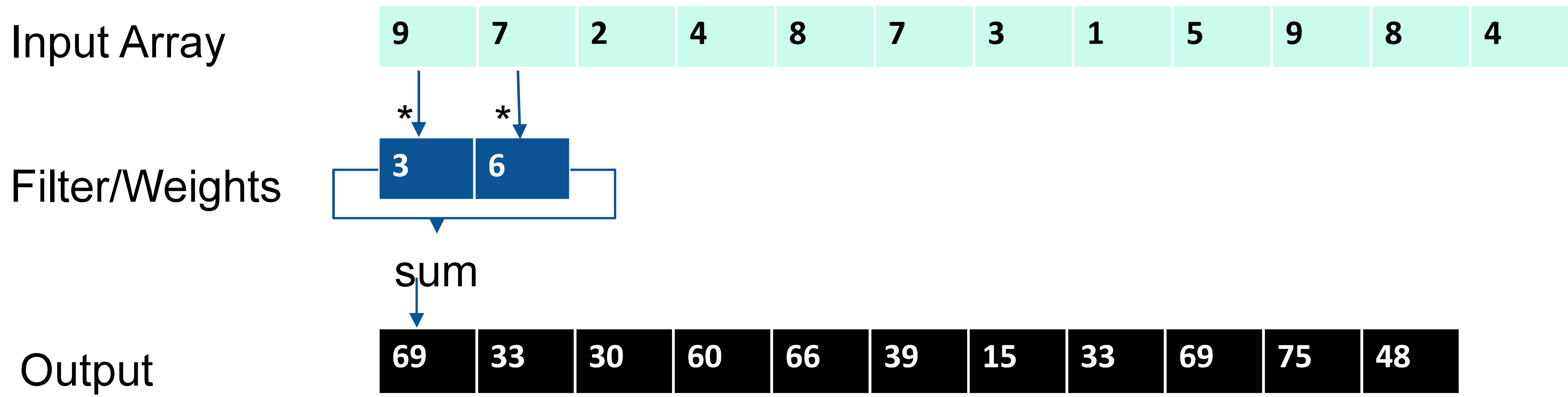


Fully-connected layer

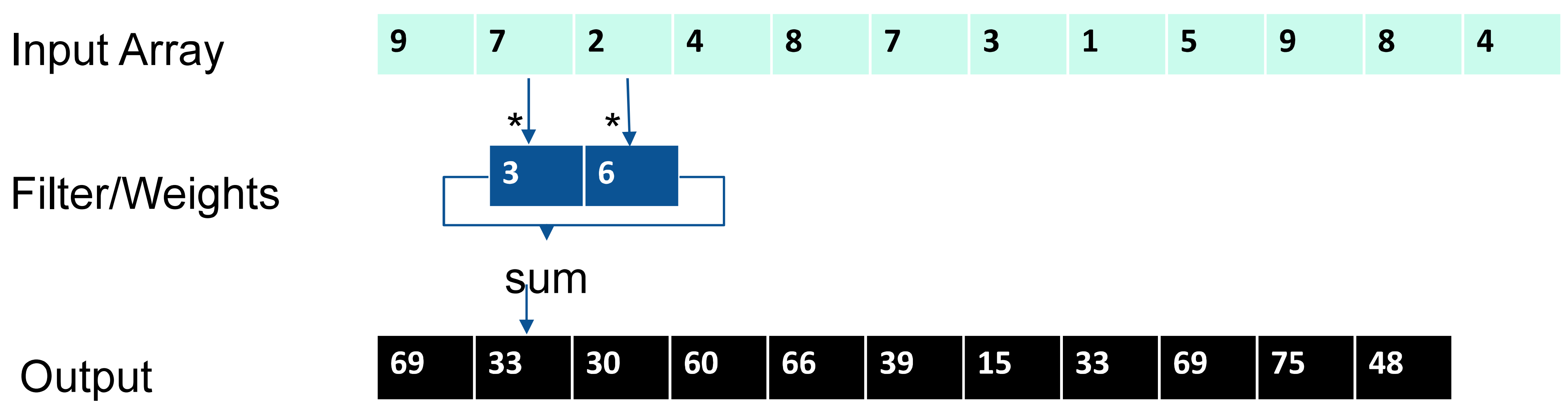


Convolutional Layer

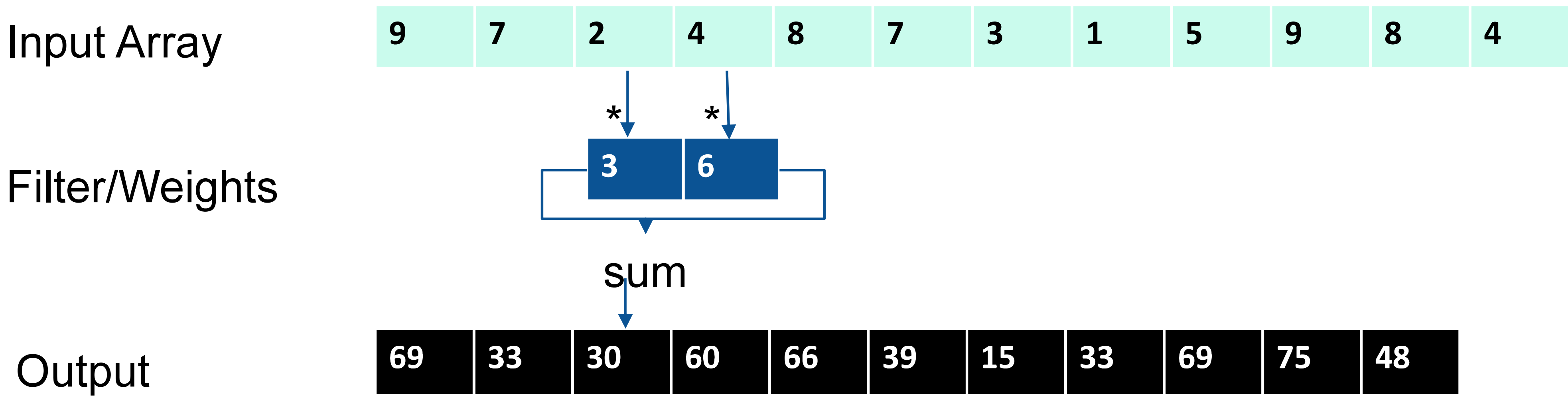
CNNs – Filter, Stride, Padding Example



CNNs – Filter, Stride, Padding Example



CNNs – Filter, Stride, Padding Example



CNNs – Filter, Stride, Padding Example

Input Array	9	7	2	4	8	7	3	1	5	9	8	4

Filter/Weights 2x1

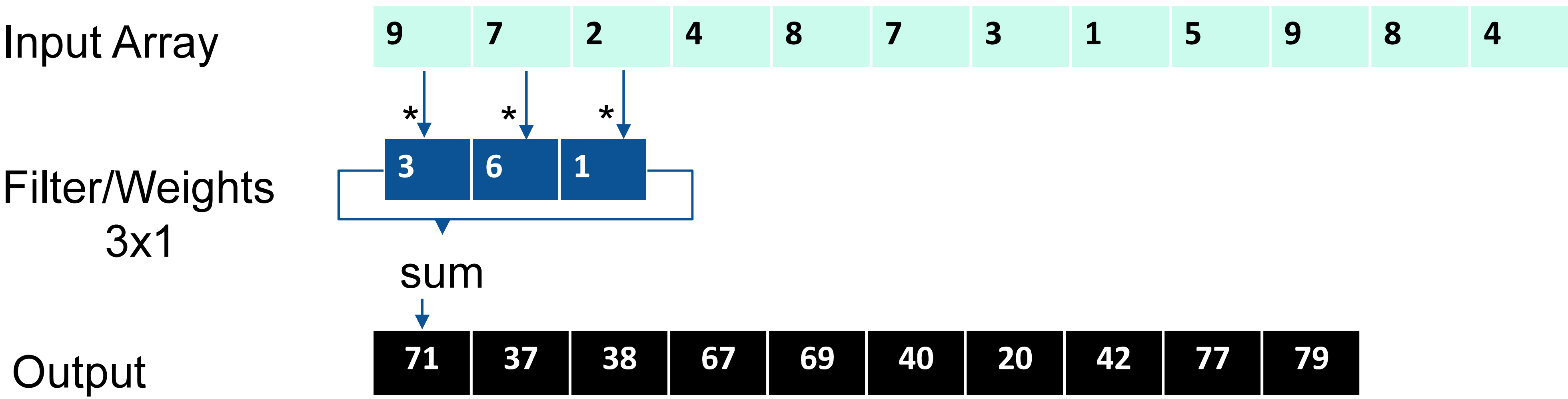
3	6

* *

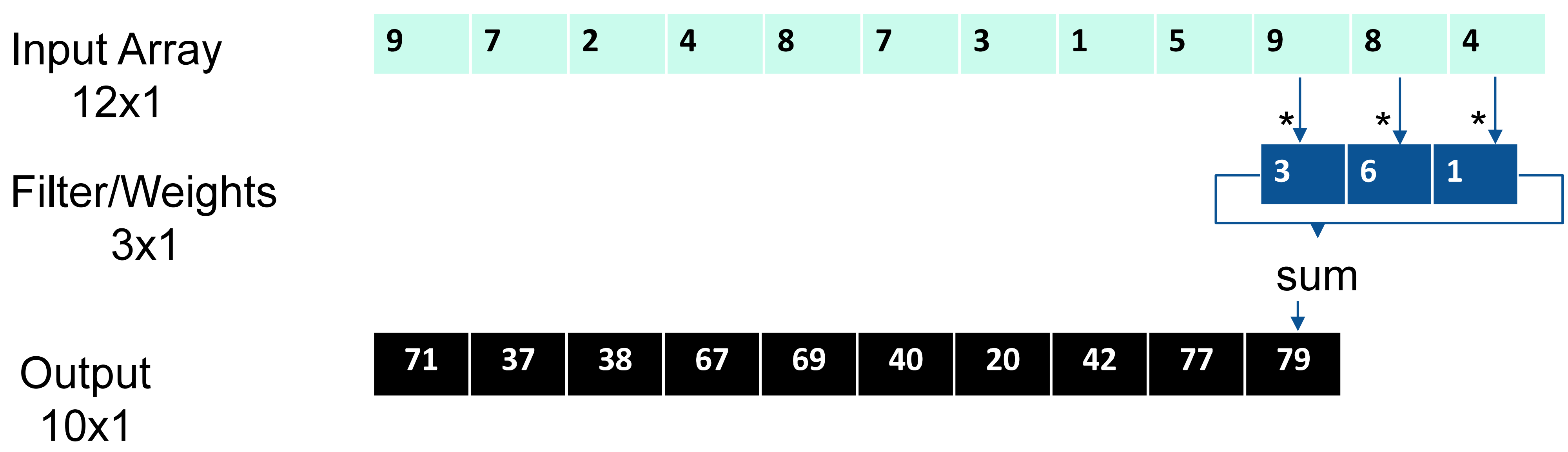
sum

Output	69	33	30	60	66	39	15	33	69	75	48

CNNs – Filter, Stride, Padding Example



CNNs – Filter, Stride, Padding Example



CNNs – Filter, Stride, Padding Example

Input Array
12x1

0	9	7	2	4	8	7	3	1	5	9	8	4	0

Filter/Weights
3x1

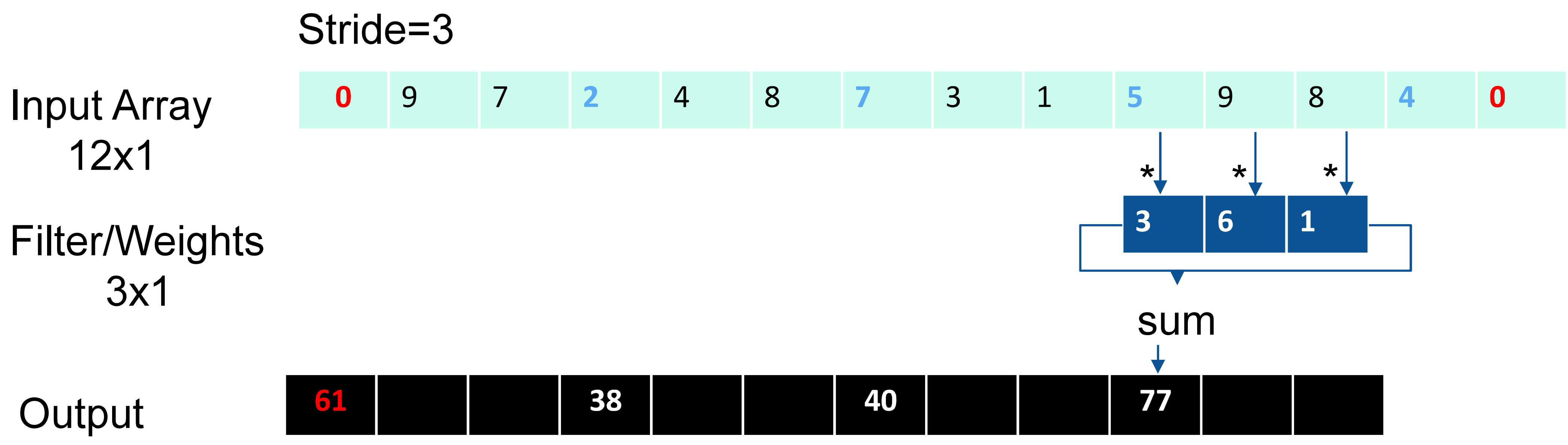
*	*	*
3	6	1

sum

Output
12x1

61	71	37	38	67	69	40	20	42	77	79	48

CNNs – Filter, Stride, Padding Example



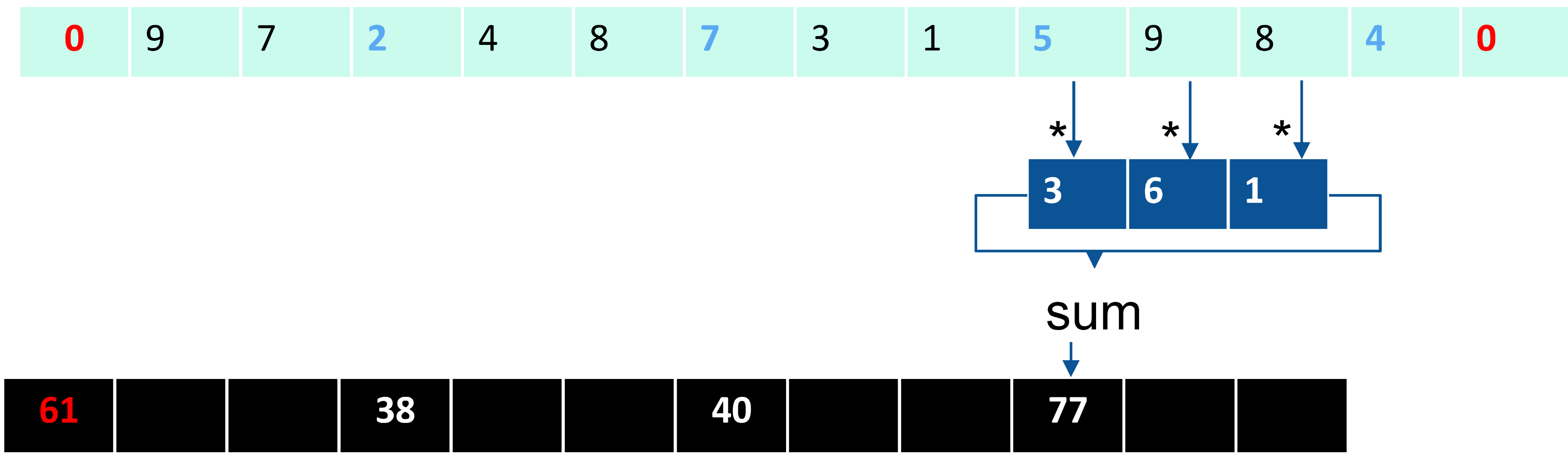
CNNs – Filter, Stride, Padding Example

Stride=3

Input Array
12x1

Filter/Weights
3x1

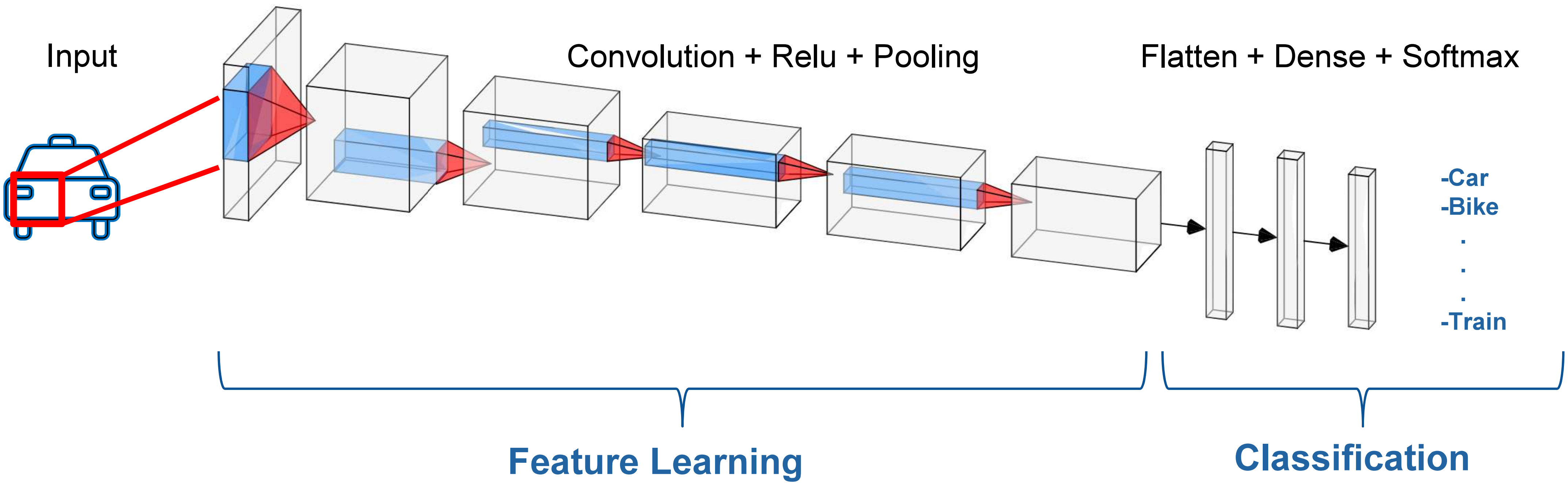
Output



Number of Output Parameters

$$(N-F+2P)/S + 1 = 4$$

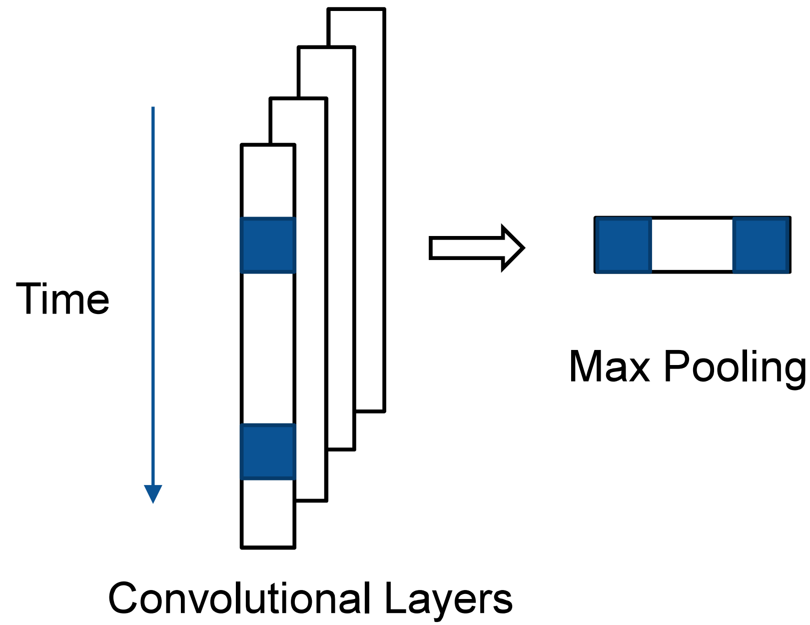
Convolutional Neural Networks



- The model learns to **extract a hierarchy of features**
- CNN has applied successfully both in 1D and 2D classification

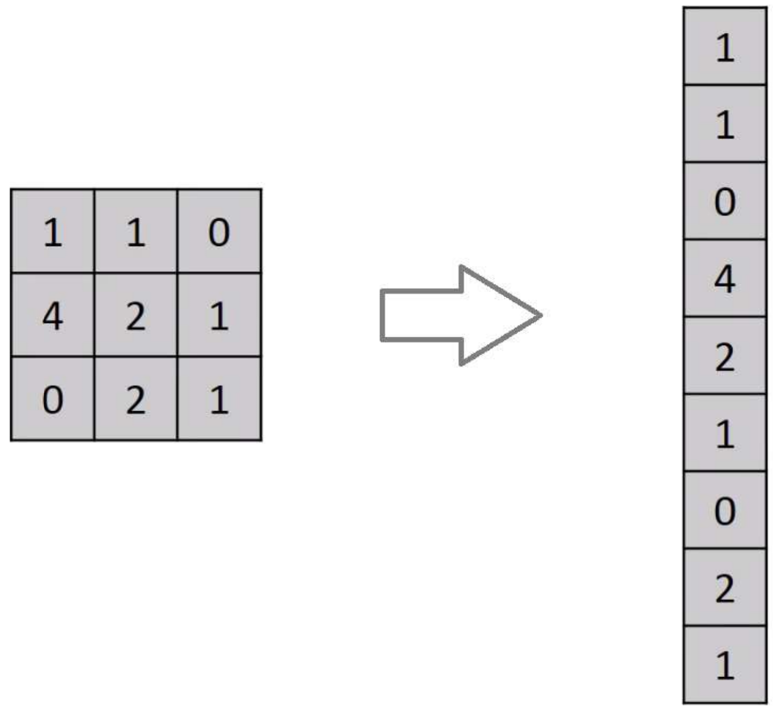
Model Layers: Pooling

- **Pooling layers** are used for **downsampling**
- It reduces the number of parameters to learn
- It reduces overfitting
- It can also result in **invariance**
  - Max pooling
  - Average pooling
  - L2 norm pooling
  - Stochastic pooling
  - Spectral pooling

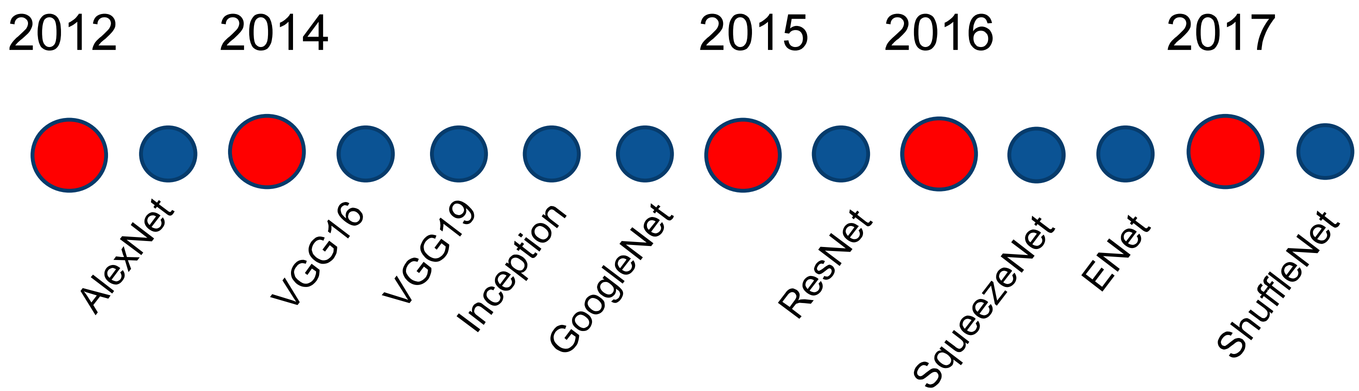


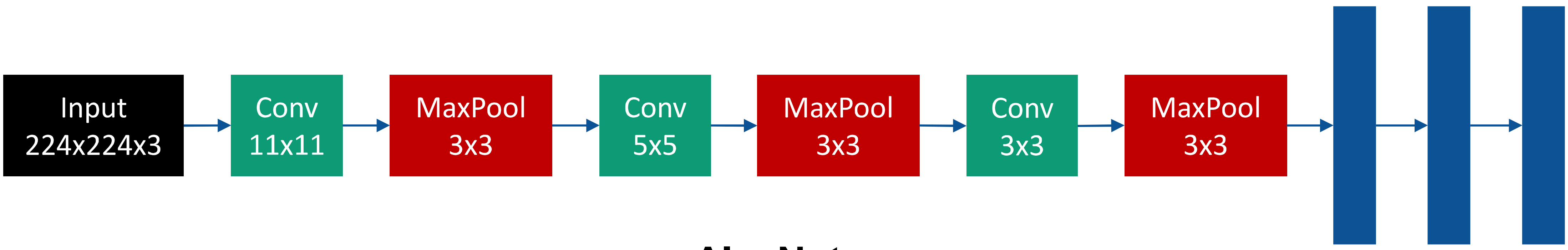
Model Layers: Flatten

- **Flattening** is converting the data into a 1-dimensional array for inputting it to the next layer.
- We **flatten the output of the convolutional layers** to create a single long feature vector.
- And it is connected to the final classification model, which is called a fully-connected layer.



Classic CNN Architectures





AlexNet

Summary

- CNNs have been proven very efficient in computer vision and it is one of the most successful application of Deep Learning
- CNNs are more efficient than a Multi-layer Perceptron architecture which is fully connected
- They exploit that fact of extracting local features in a hierarchical way
- Training in CNN can be more challenging than Multi-layer Perceptron

References

- Ravi et al. Deep Learning for Health Informatics, IEEE Journal of Biomedical and Health Informatics, 21(1), 2017
- Kamath, Deep Learning for NLP Applications, Springer, 2019
- Foster, Generative Deep Learning – Teaching Machines to Paint, Write, Compose and Play, O'Reilly, 2019
- https://en.wikipedia.org/wiki/Convolution