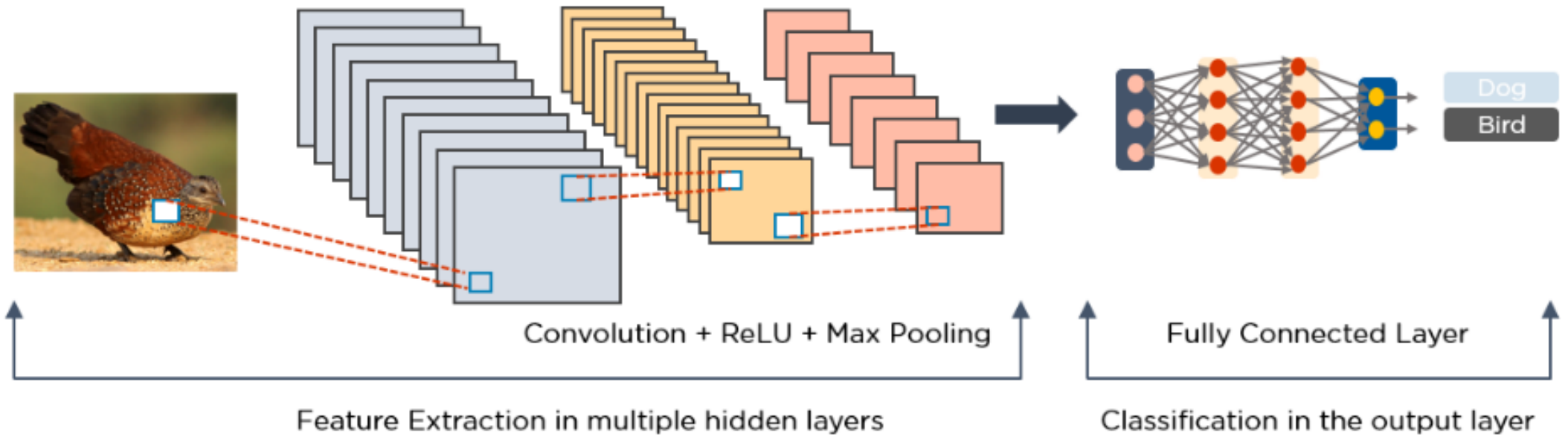
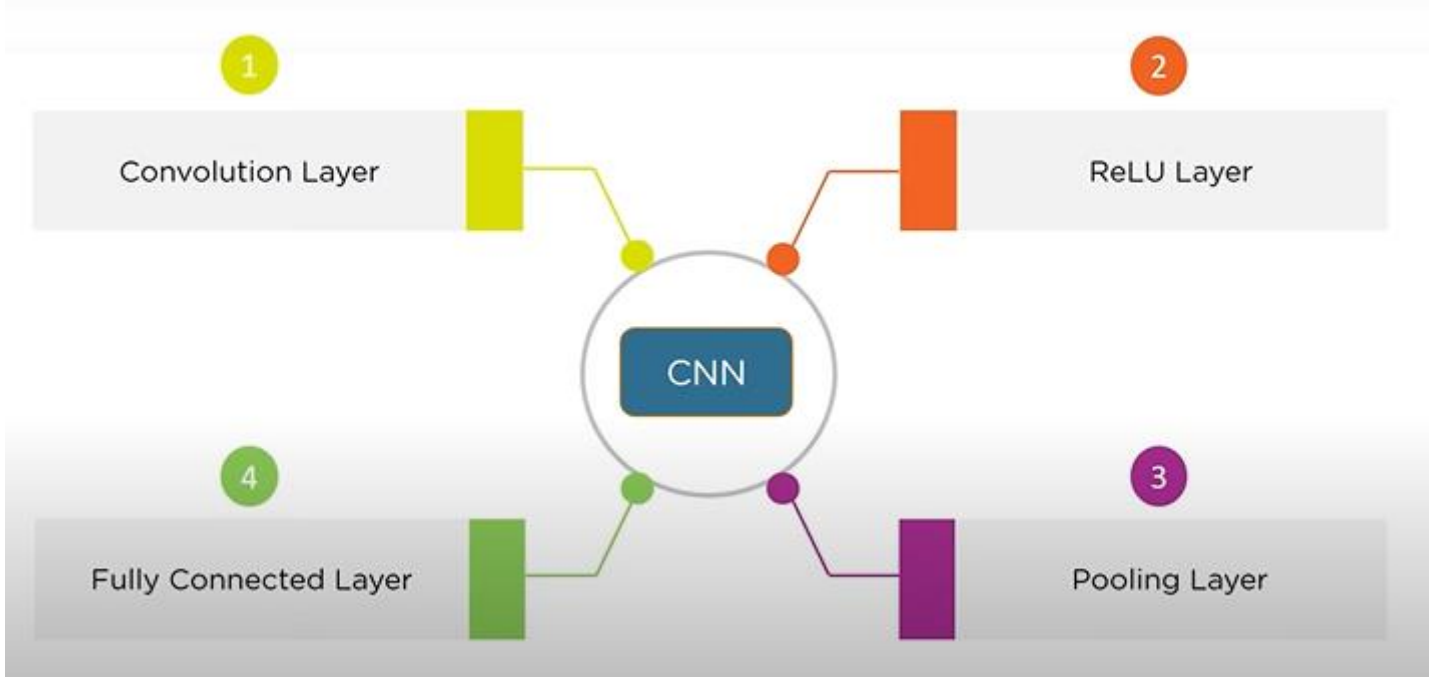
Convolutional Neural Networks (CNNs)



CNN's, also known as ConvNets, consist of multiple layers and are mainly used for image processing and object detection.

CNN Layers



There are four layers in CNN:

1.Convolutional Layer - ***CNN has a convolution layer that has several filters to perform the convolution operation .***

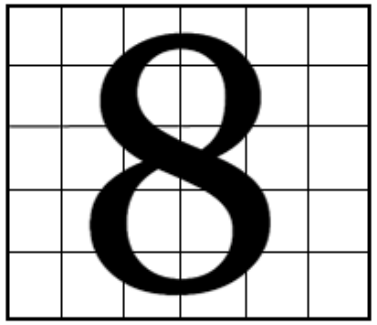
2.ReLU Layer - it brings non-linearity to the network and converts all the negative pixels to zero. The output is a rectified feature map.

3.Pooling Layer - pooling is a down-sampling operation that reduces the dimensionality of the feature map.
The pooling layer then converts the resulting two-dimensional arrays from the pooled feature map into a single, long, continuous, linear vector by flattening it.

4. Fully Connected Layer - this layer recognizes and classifies the objects in the image.

In CNN, every image is represented in the form of an array of pixel values

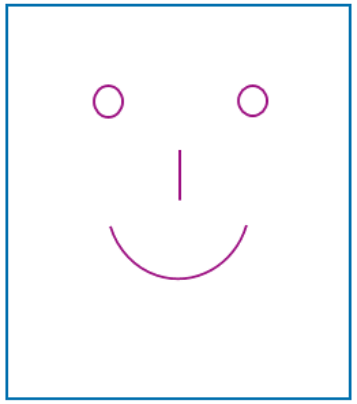
Real Image of the digit 8



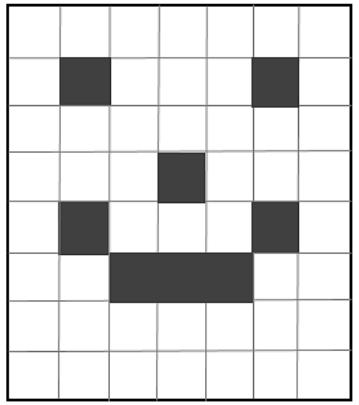
Represented in the form of an array

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

Digit 8 represented in the form of pixels of 0's and 1's



Real Image



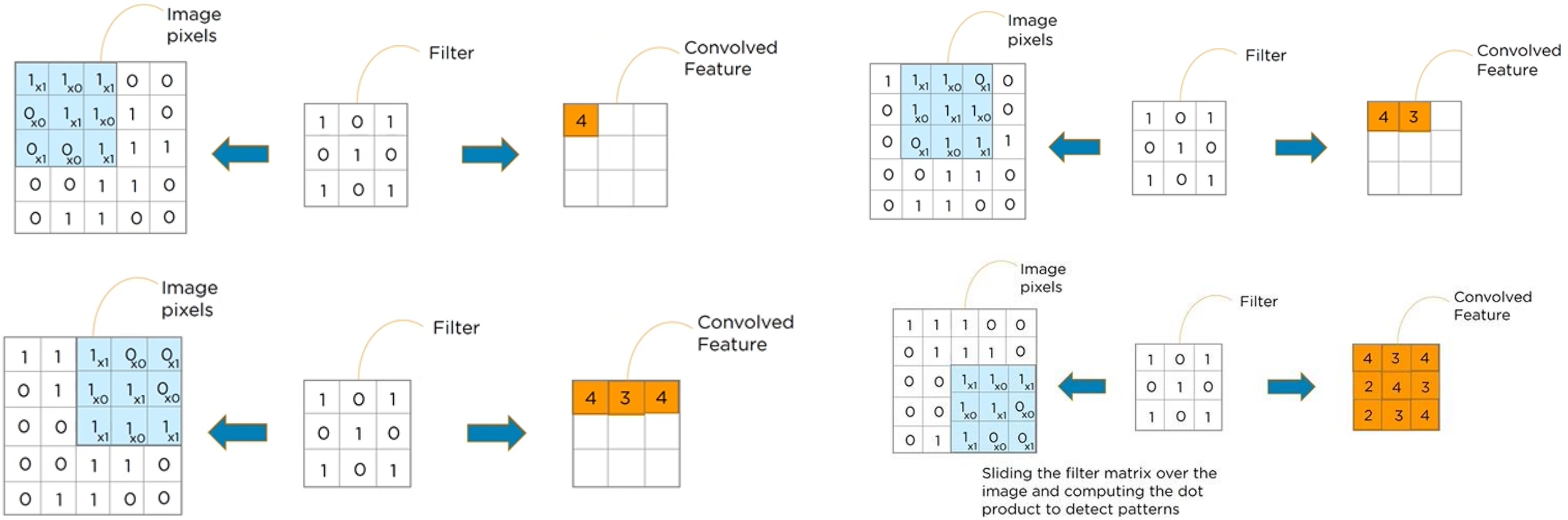
Represented in the form of black and white pixels

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
0	0	0	0	0	0	0

Image represented in the form of a matrix of numbers

Feature map Extraction

Filters – like a weight matrix with which we multiply a part of the input image to generate a convoluted output.
Assume we have an image of size 28*28. Here filter is a 3*3 matrix which is multiplied with each 3*3 section of the image to form the convolved feature.
The filter values are updated like weight values during back propagation for cost minimization.



Convolution operation using two matrices, a and b, of 1 dimension

a * b

Multiply the arrays element wise

Sum the product

[5, 6, 6]

17

a = [5, 3, 2, 5, 9, 7]

b = [1, 2, 3]

a * b

Multiply the arrays element wise

Sum the product

[5, 6, 6]

[3, 4, 15]

17

22

a = [5, 3, 2, 5, 9, 7]

b = [1, 2, 3]

Let's understand the convolution operation using 2 matrices a and b of 1 dimension

a * b = [17, 22]

a = [5, 3, 7, 5, 9, 7]

b = [1, 2, 3]

Matrix a and b

Convolution

a * b

Multiply the arrays element wise

Sum the product

a = [5, 3, 2, 5, 9, 7]

b = [1, 2, 3]

[5, 6, 6]

[3, 4, 15]

[2, 10, 27]

17

22

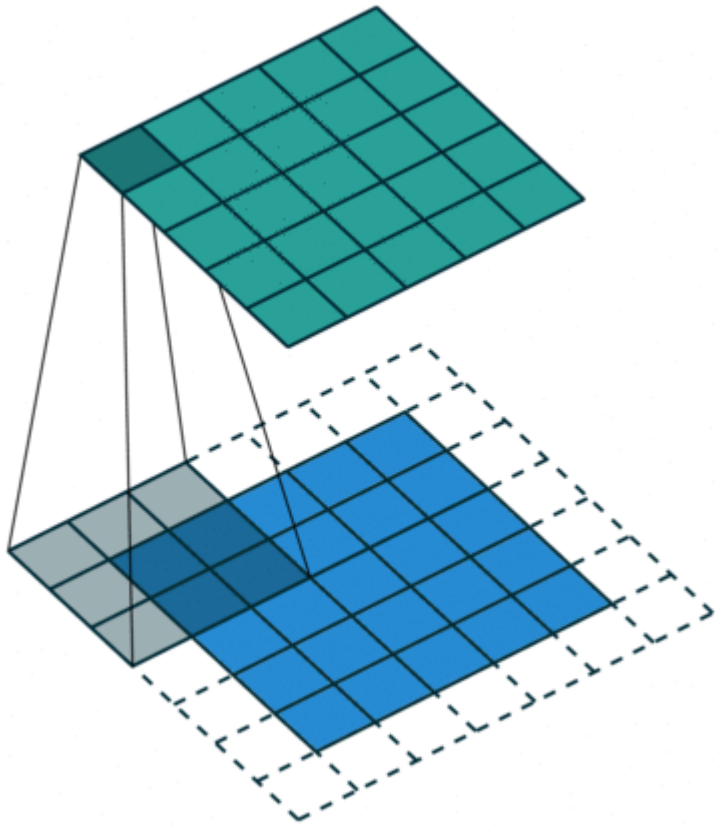
39

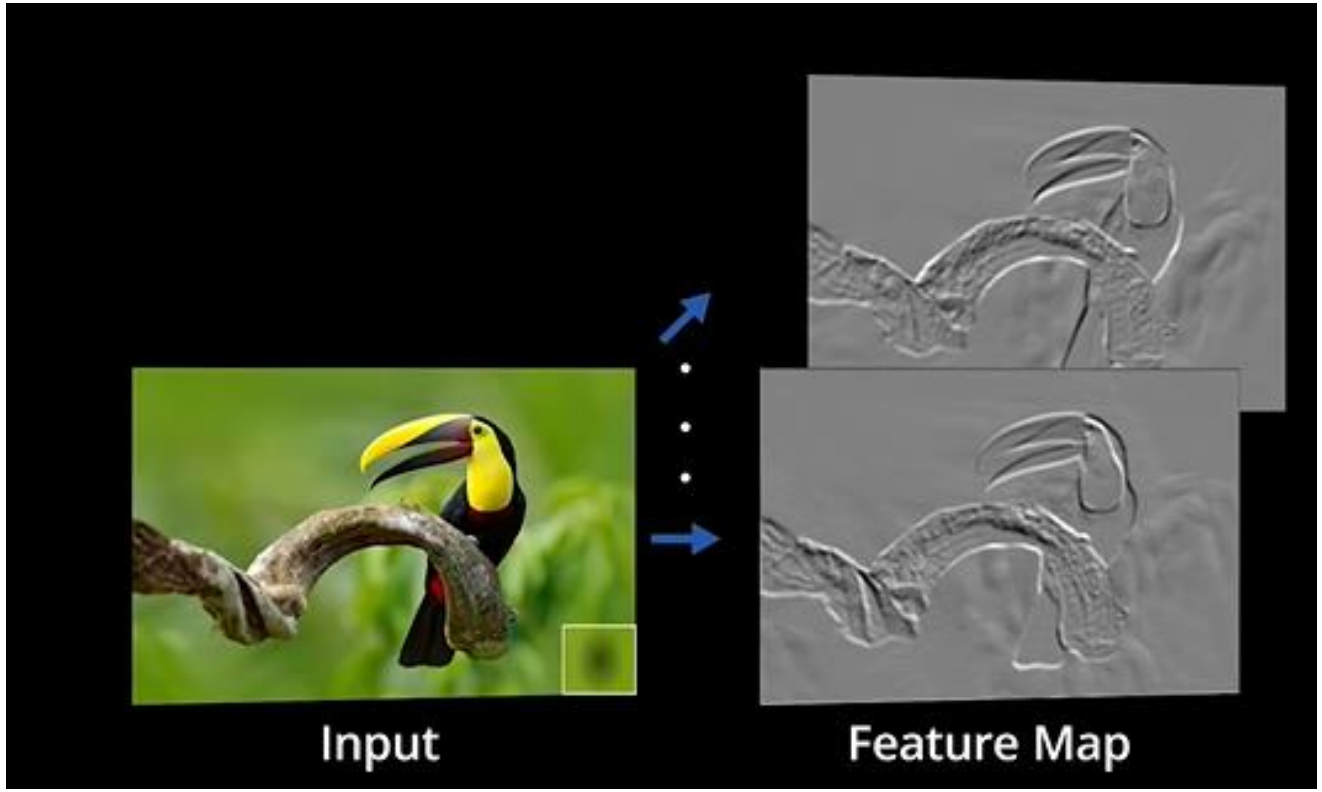
a * b = [17, 22, 39,.....]

Convolution between two functions in mathematics produces a third function expressing how the shape of one function is modified by other

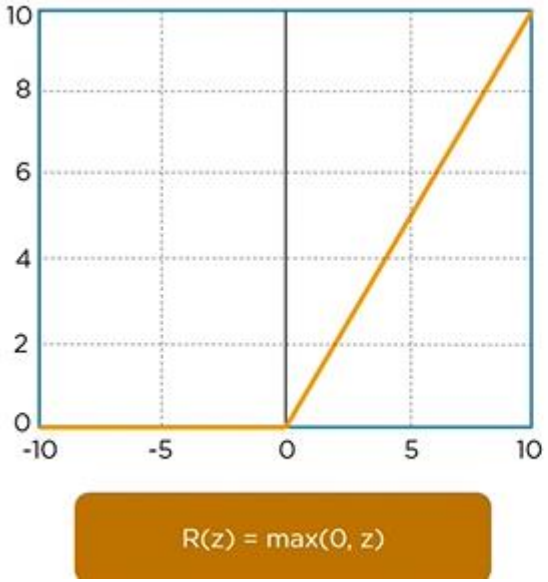
Original	Gaussian Blur	Sharpen	Edge Detection
$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$
			

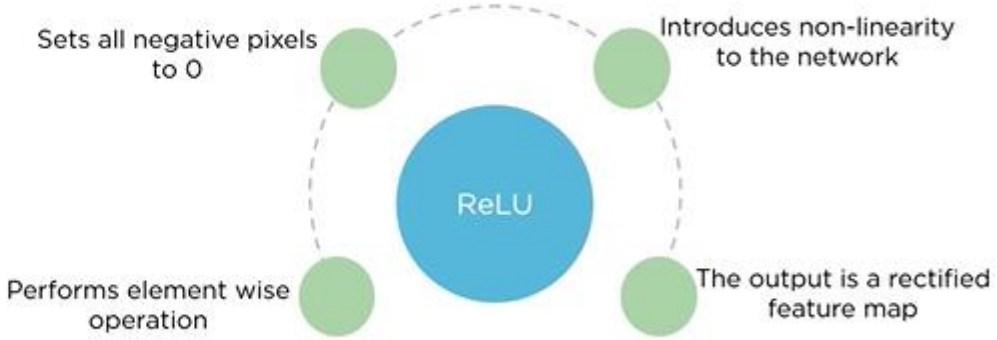
CNN and Feature Mapping

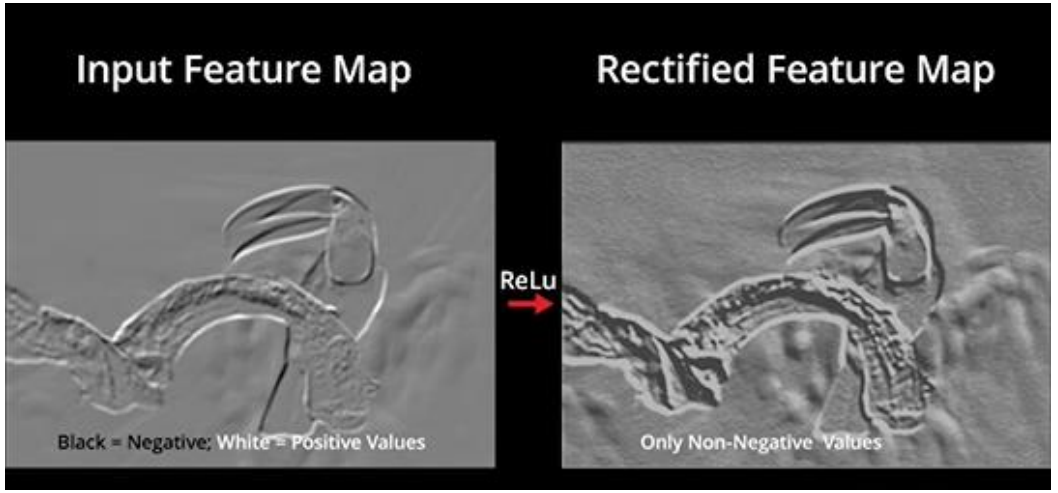




ReLu Layer


10
8
6
4
2
0
-10
-5
0
5
10
R(z) = max(0, z)

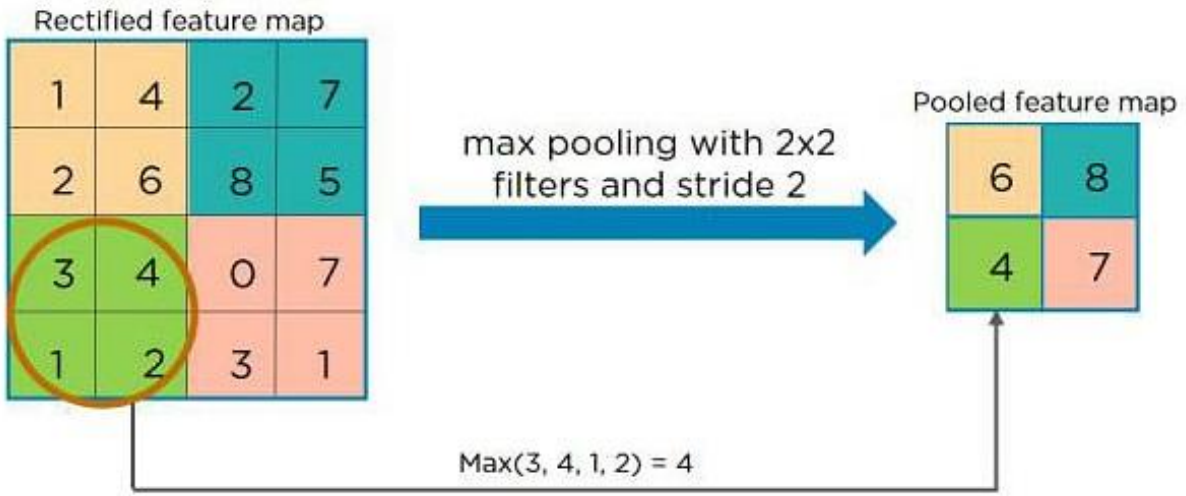

Sets all negative pixels to 0
Introduces non-linearity to the network
ReLU
Performs element wise operation
The output is a rectified feature map


Input Feature Map
Rectified Feature Map
ReLu
Black = Negative; White = Positive Values
Only Non-Negative Values

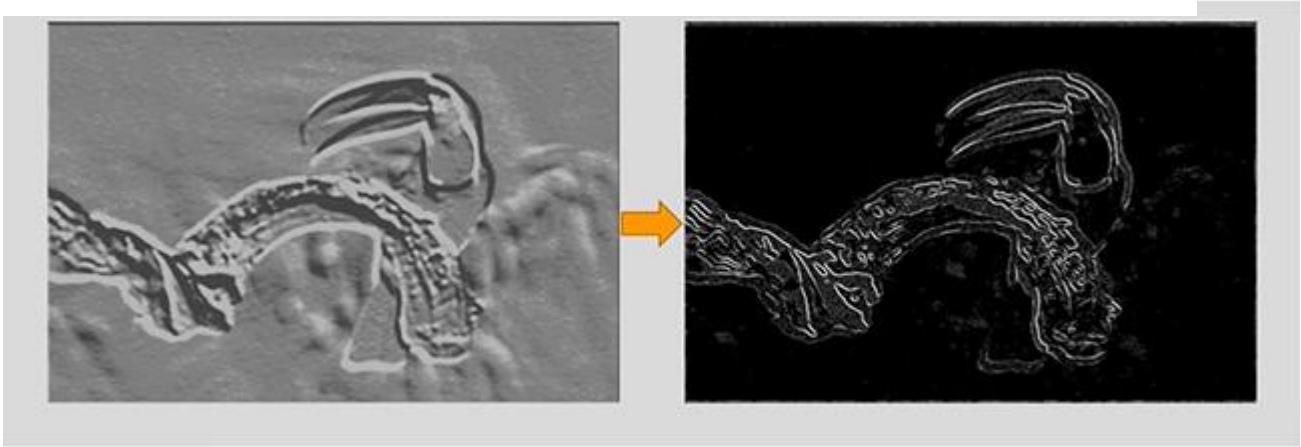
Pooling

Pooling

- Used to reduce the spatial dimensions of a CNN
- Performs down-sampling operation to reduce the dimensionality
- Creates a pooled feature map by sliding a filter matrix over the input matrix

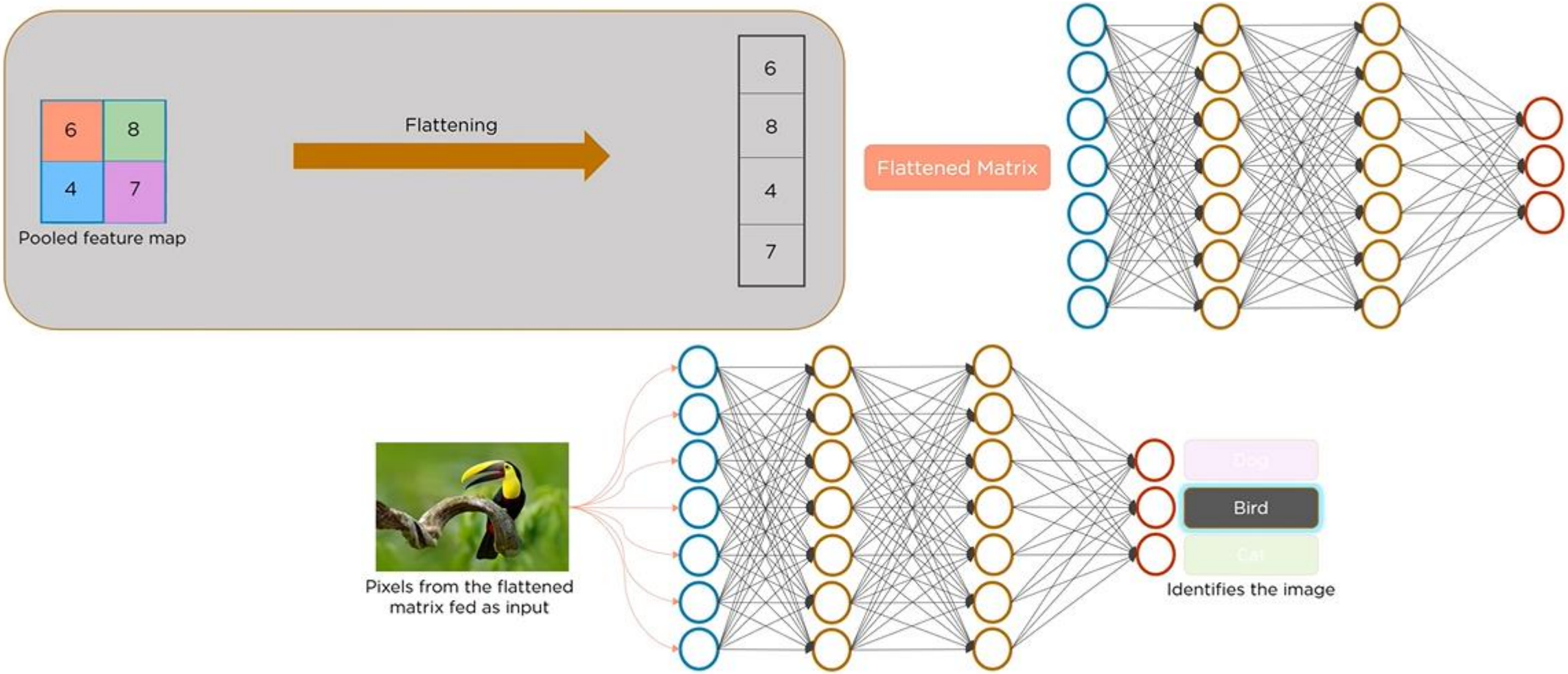


To reduce a number of parameters and prevent over-fitting.
Type of pooling is a pooling layer of filter size(2,2) using the MAX operation, Average pooling

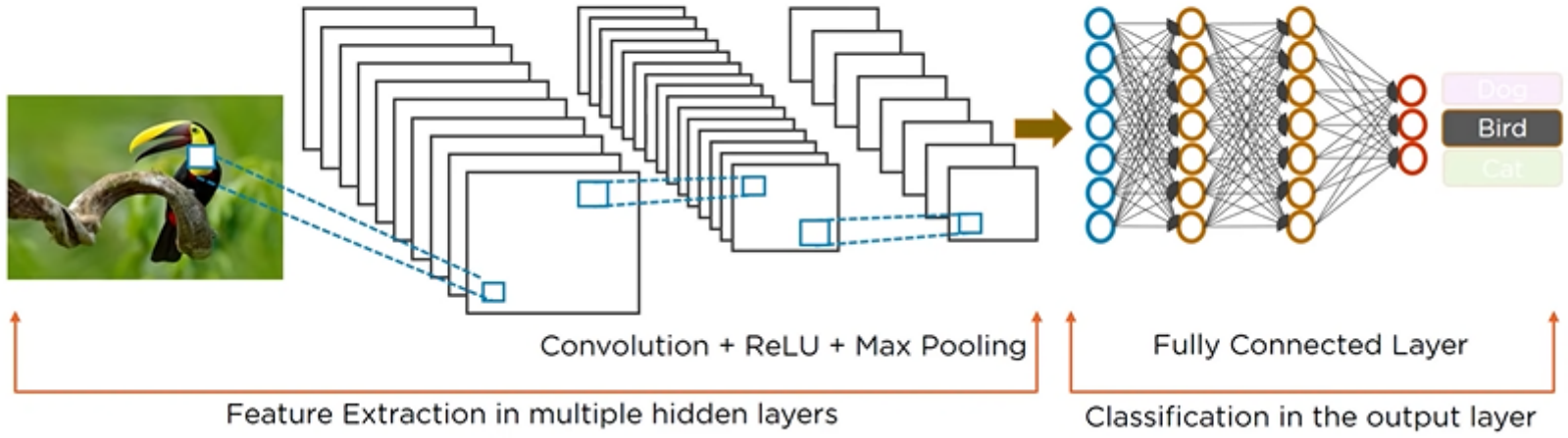


Identifies the edges, corners and other features of the bird

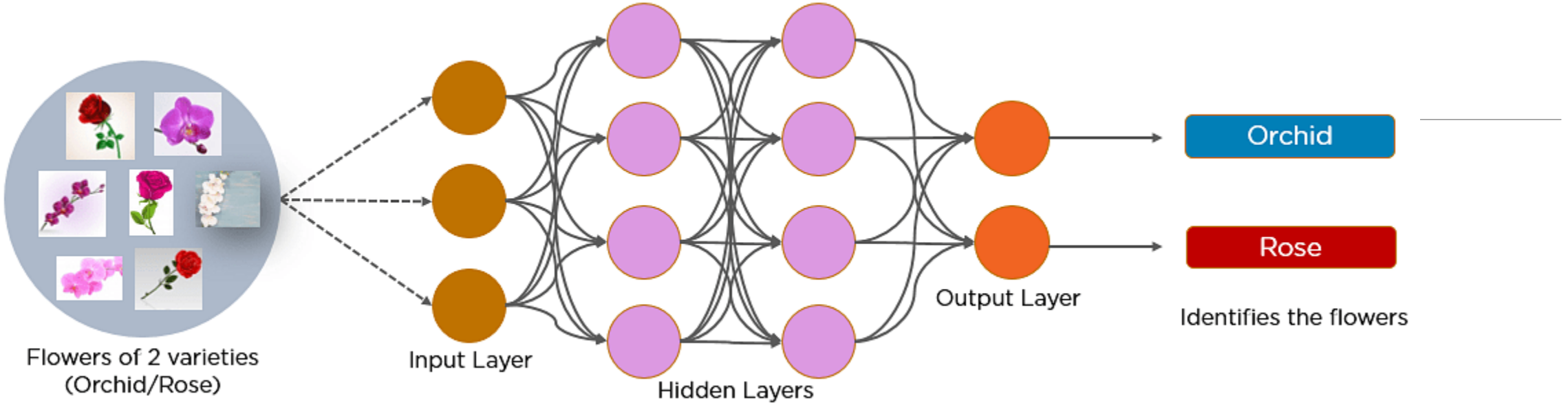
Flattering

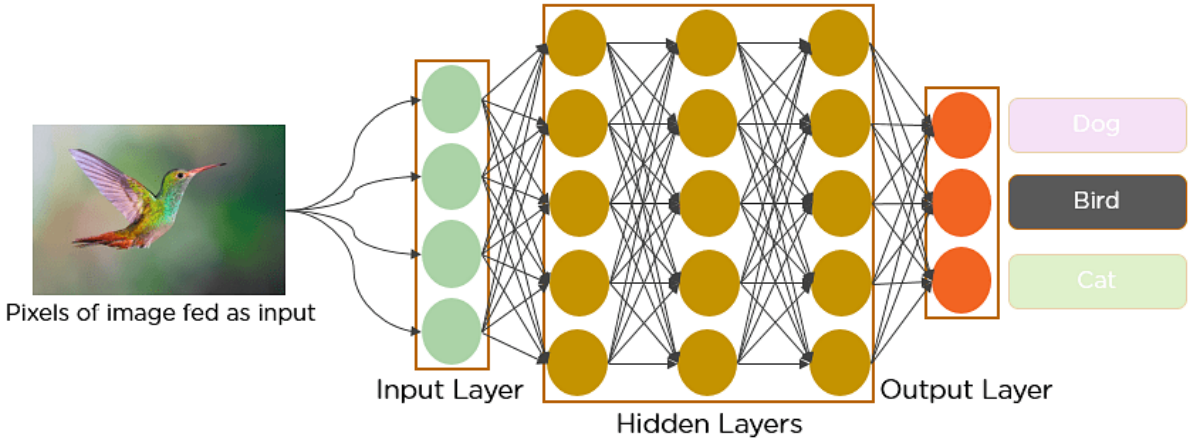

6
8
4
7
Pooled feature map
Flattening
6
8
4
7
Flattened Matrix
Pixels from the flattened matrix fed as input
Bird
Identifies the image

CNN - Summary


Dog
Bird
Cat
Convolution + ReLU + Max Pooling
Fully Connected Layer
Feature Extraction in multiple hidden layers
Classification in the output layer

CNN- Applications





To identify satellite images, process medical images, forecast time series, and detect anomalies