

One-Hot Label
0001000000

Model Prediction 1: 000.10000.90
Model Prediction 2: 000.90000.10

loss = $-\log(.1) = 2.30$

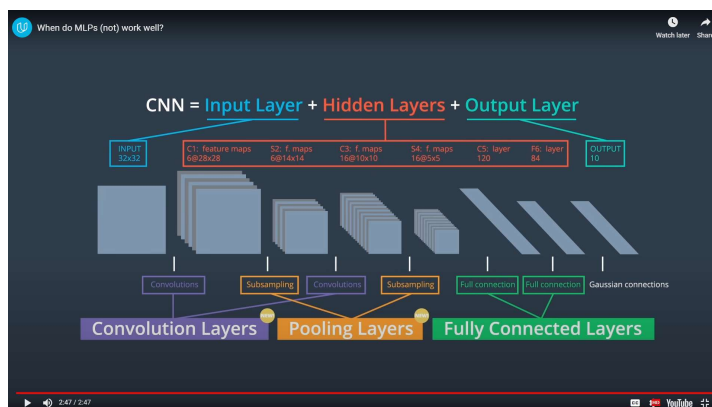
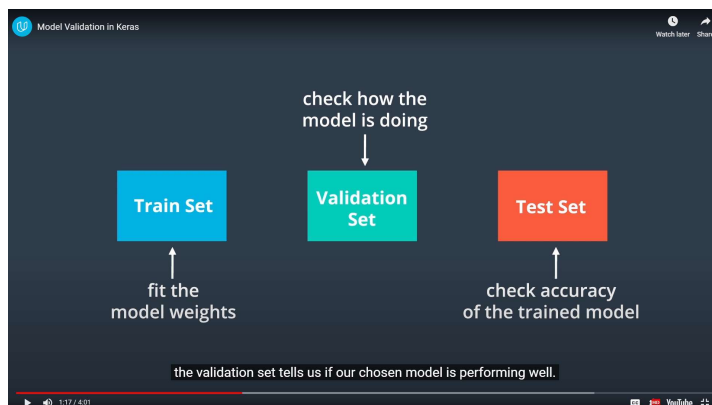
where it changes to being 90% sure that the image depicts a three,

这里，一个 vector 是一个 label，不是一个位置上的 (0, 1) 是一个 label!!!

Categorical Cross Entropy

LOSS = $\begin{cases} \text{LOWER :)} & \text{When label and prediction agree} \\ \text{HIGHER :(} & \text{When label and prediction disagree} \end{cases}$

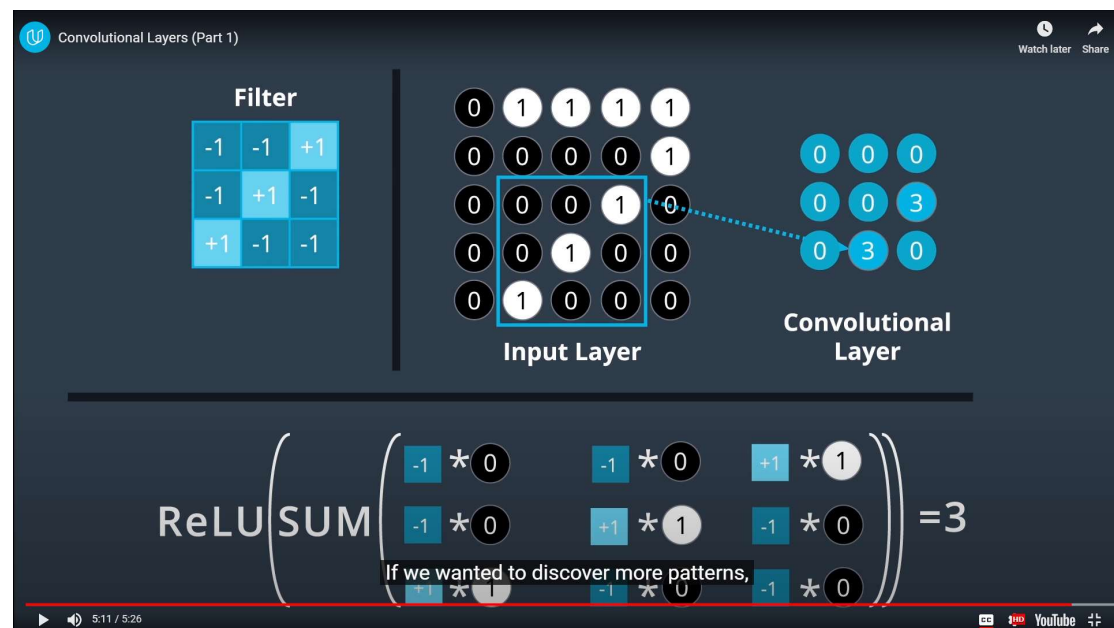
We want its predictions to agree with the label.



CNN

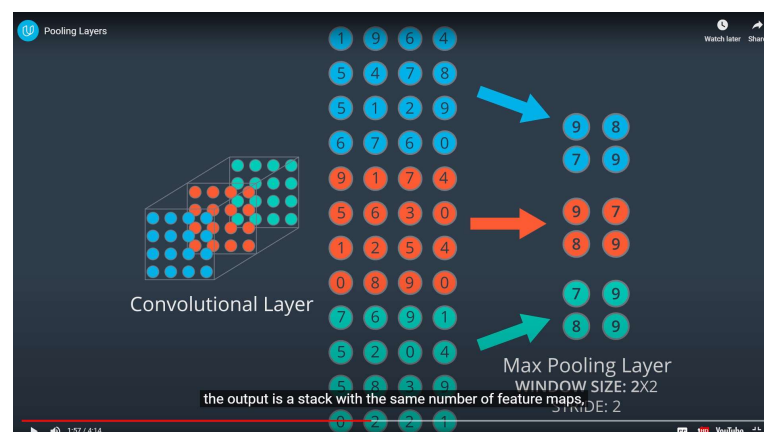
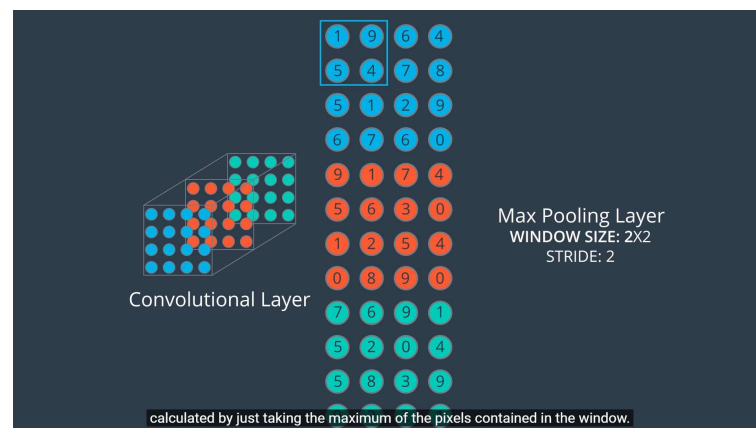
Filter is used to detect a certain pattern.

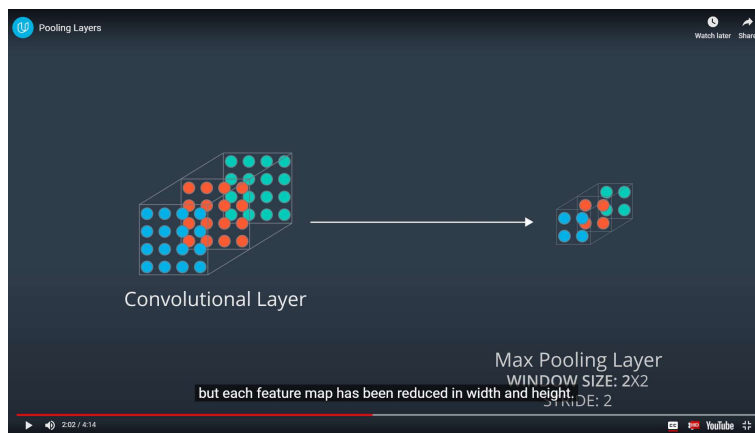
In the picture, the max value can get is 3. It tells that the region in convolutional layer which has 3 matches the filter pattern.



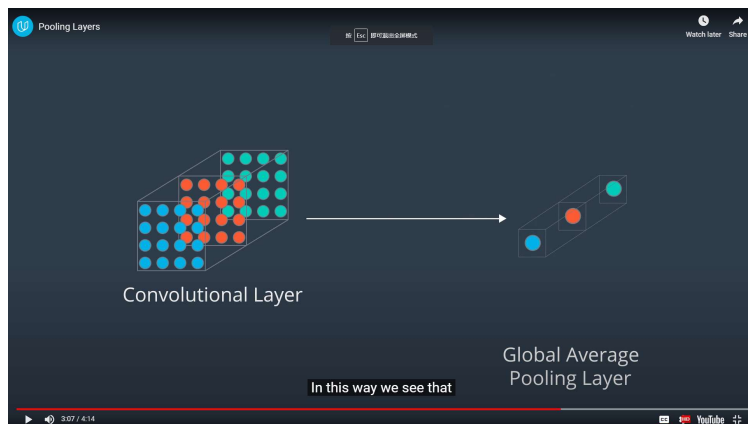
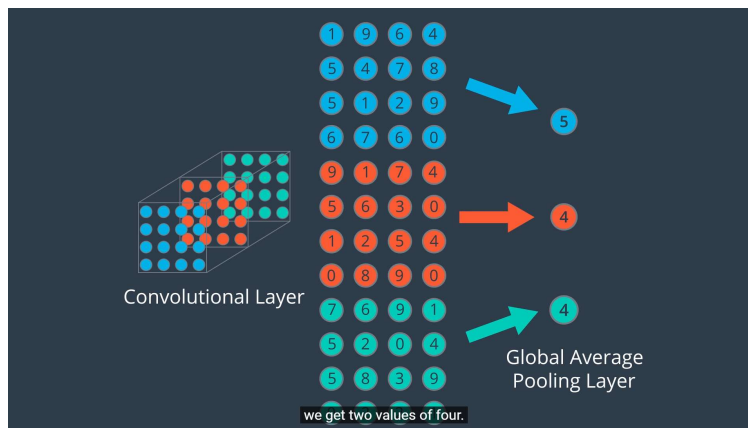
Pooling layer

maxpooling





Global average pooling



Construct a CNN

CNN19 CNNs For Image Classification

MaxPooling2D(pool_size, stride, padding)

Watch later Share

INPUT

CONV

POOL

CONV

POOL

CONV

POOL

2 2 2 2

(default ok)

Let's double check this by entering the code.

5:20 / 8:57

YouTube

CNN19 CNNs For Image Classification

Convolution2D(filters, kernel_size, strides, padding, activation, input_shape)

Watch later Share

INPUT

CONV

CONV

CONV

16 2 1 'same' 'relu' (32, 32, 3)

32 2 1 'same' 'relu'

64 2 1 'same' 'relu'

And we'll use the 'relu' activation function in all of our convolutional layers.

4:15 / 8:57

YouTube