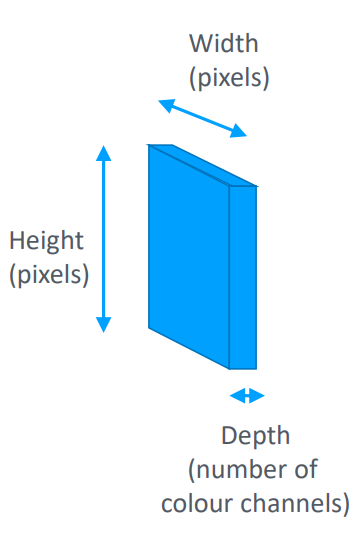
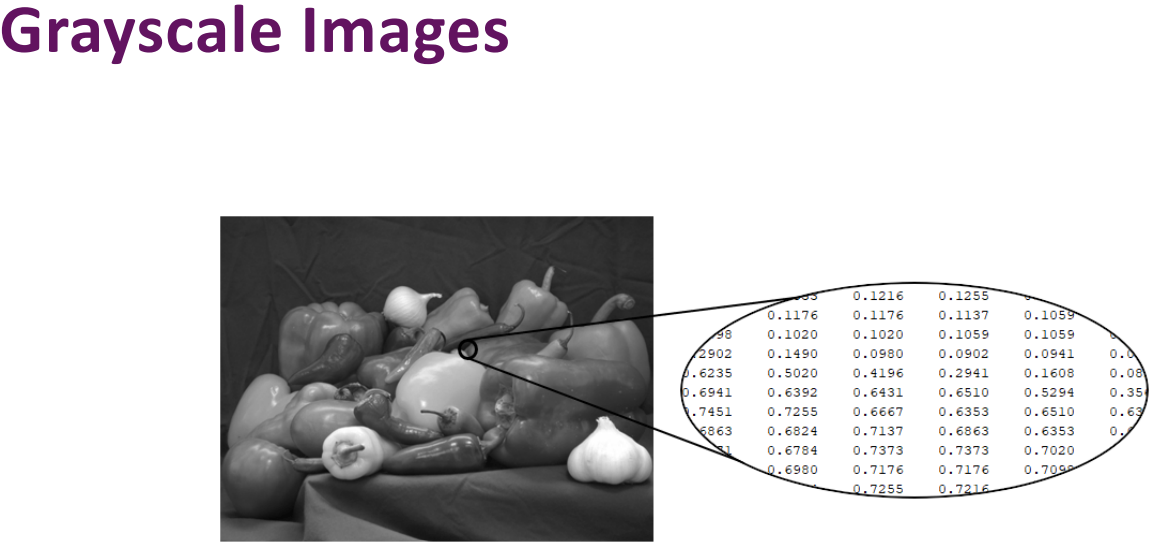
12/03/24 **Lecture 7: CNN**

A convolutional network input is a three-dimensional array of numbers, with the first two dimensions representing the image's dimensions in pixels and the third dimension representing the channel dimension.



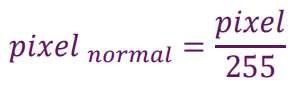
**Image types**:





**Normalisation**:

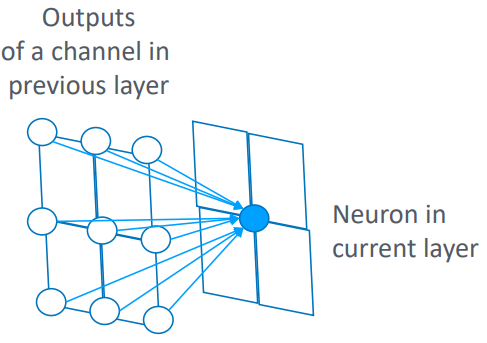
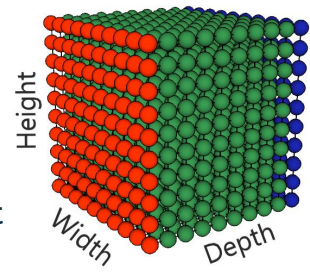
Image pixel values range from 0 to 255. Optimisers for DNNs can scale data into a floating range between 0.0 and 1.0, or normalise it by shifting and scaling the data.



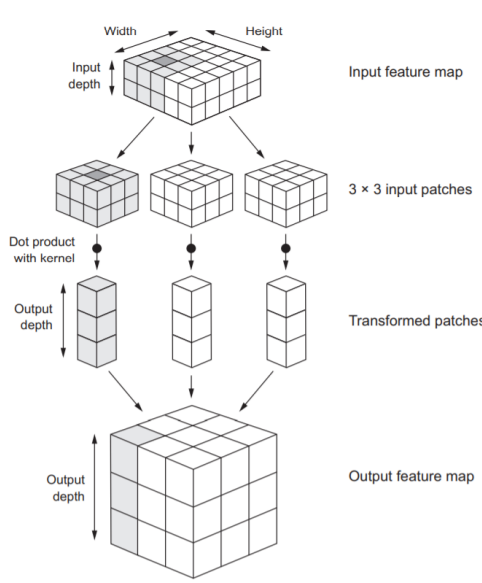
A smaller image is better (More computational/memory efficient)

**Convolutional layer**

A convolutional layer is a 3-dimensional grid of neurons operating over 3D tensors called feature maps. It has width, height, and depth, representing the input image's sensitivity and the different filters applied to it. Convolutions with larger footprints are typical in practice, with simple 2 by 2 convolutions illustrated.



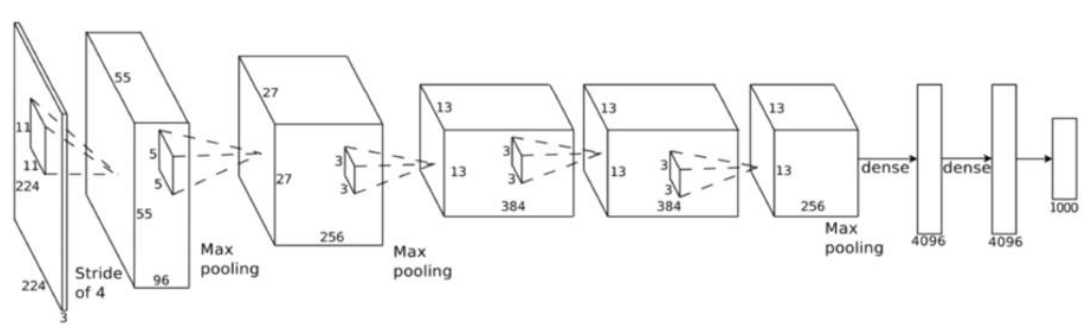
Convolution layers divide an input feature map into patches using a filter window, transforming each patch into a 1D vector using the convolution kernel, resulting in 3D tensors:



**Activation function**

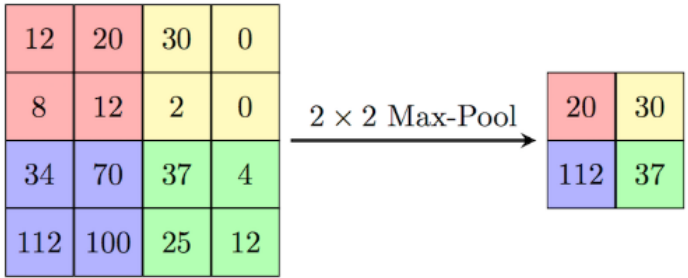
Softmax - In a classifier using neural networks, the final output layer typically has N outputs, each with a probability of belonging to one of the N categories. To interpret these outputs, the N raw outputs are often fed to the softmax function.

**Architecture example**



**Max pooling**

Input dimensionality reduction:



**Strides**:

