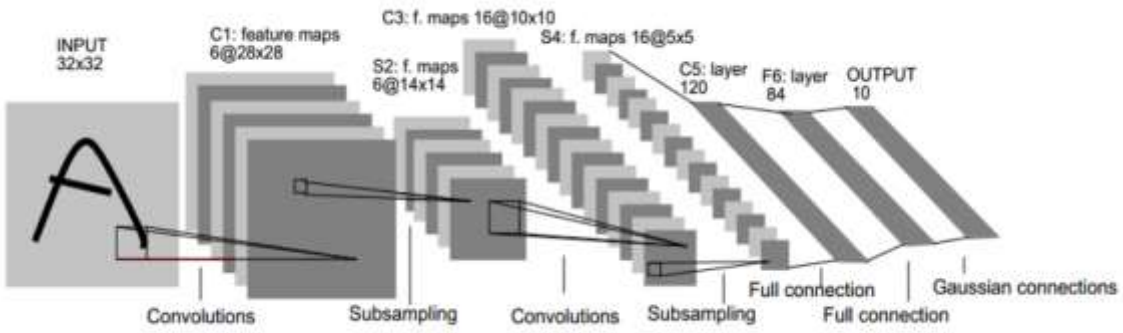
What is CNN?

Feng Su
Fs2658

"In deep learning, a convolutional neural network(CNN or ConvNet) is a class of deep neural network, most commonly applied to analyzing visual imagery "
*(In this report, I will just talk about Forward Propagation)

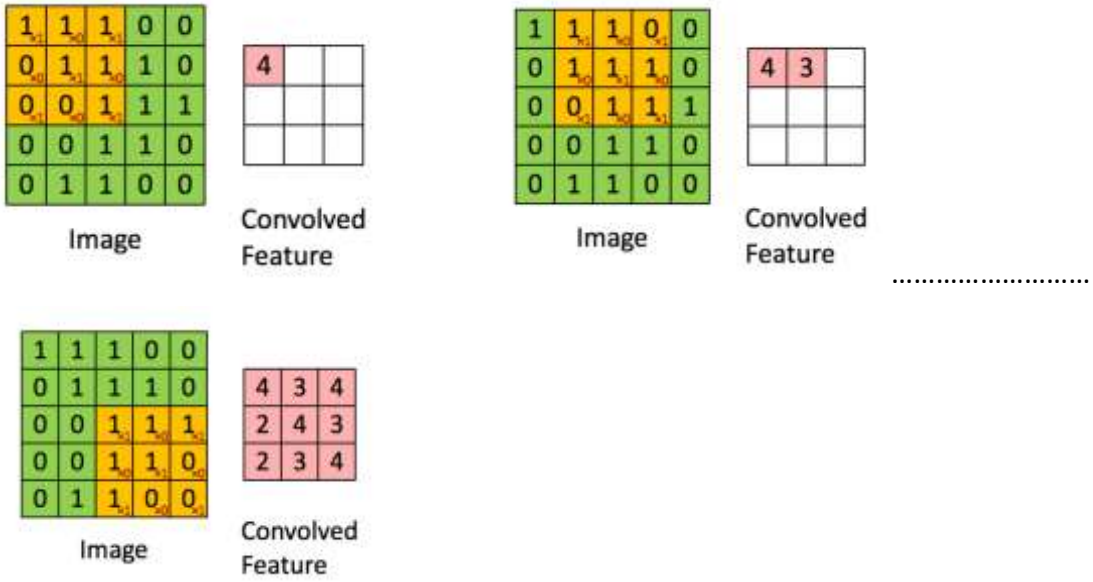
The main architecture:



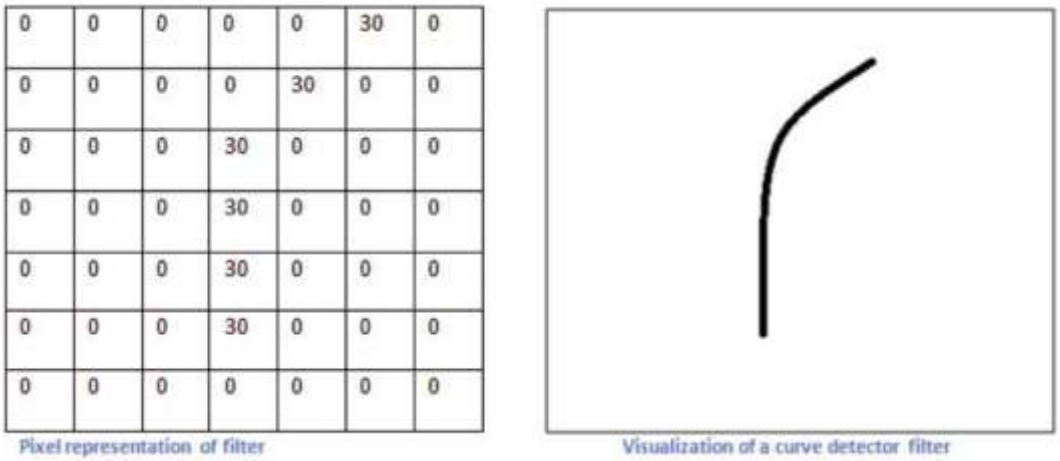
(figure 1)
Before I dig into the main architecture, I will briefly introduce those main parameters we used in this neural network.

- Input: Those images waiting to be convolved which including the image height, width and depth
- Filter and kernel: a small matrix used for features detection. Specifying the width and height of the 2D convolution window. K kernels waiting to be applied to the image, each kernel is convolved with input volumn and the output of each convolution.
- Activation function: used to increase non-linearity which can help the model to learn more complicated features (RELU used most ofen)

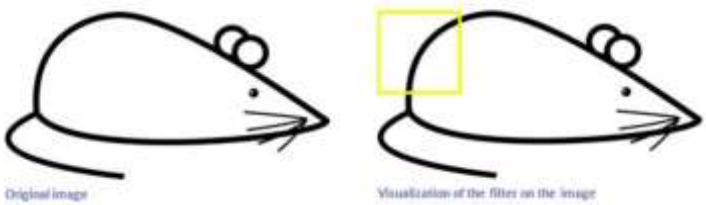
Now, let's start to look at the architecture itself.
(1) The Convolution layers:
In general, convolutional layers apply a convolution operation to the input, passing the result to the next layer.
From my personal understanding, at the very beginning of the convolutional process, we need first apply the filter(kernel) to process the image. A very classic way to do this is use the surrounding 8 features with the kernel to generate the convolved features. For example, we have a 5*5 image and a kernel with size 3*3 and the convolution process can be stated as following:



And this process is what we called convolution. So what is the reasoning behind this convolutional step? In my personal opinion, every convolution step helps our model to learn some specific features regard to image itself. For example, if we want to detect a specific line or curve in a single image, we should design a filter(kernel) which can have a very high output(the ability to distinguish this feature) to this kind of line or curve but have a really low output to other features.
Example:



When we move this filter across the original image



Visualization of the receptive field

0	0	0	0	0	0	30
0	0	0	0	50	50	50
0	0	0	20	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0
0	0	0	50	50	0	0

Pixel representation of the receptive field

*

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

Multiplication and Summation = (50*30)+(50*30)+(50*30)+(20*30)+(50*30) = 6600 (A large number!)

Visualization of the filter on the image

0	0	0	0	0	0	0
0	40	0	0	0	0	0
40	0	40	0	0	0	0
40	20	0	0	0	0	0
0	50	0	0	0	0	0
0	0	50	0	0	0	0
25	25	0	50	0	0	0

Pixel representation of receptive field

*

0	0	0	0	0	30	0
0	0	0	0	30	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	30	0	0	0
0	0	0	0	0	0	0

Pixel representation of filter

Multiplication and Summation = 0

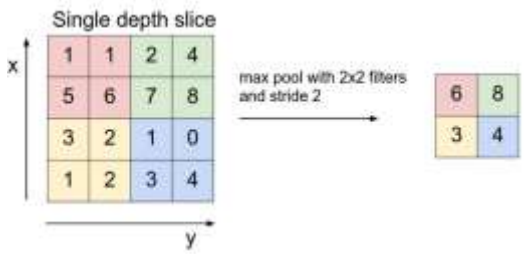
We can see that when the filter moved to the feature area it generates a very large output while other area is very small and by this we can conclude that our model obtained some information about this specific feature. And we can use different convolutional with different number and kinds of filter (kernel) to do more learning process.

(2) Pooling layer:

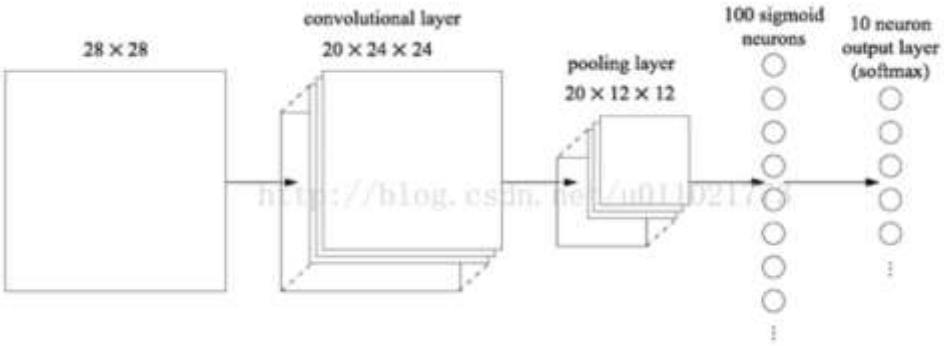
From figure one we can see that there is a subsampling between convolution layers and this is exactly what happens in pooling layers.

In general, Pooling layers are used to reduce the input space and thus complexity and processing time.

For example, we take a small area of those output images of previous convolutional layers, we randomly choose a 5*5 matrix and if we are apply the maxing pooling method then we choose the biggest pixel among those 25 pixels. If we are apply the average pooling method then we choose the average of those 25 pixels. This process helps us shrink the output image size.



(3) The Flatten and Full connected layer:



The flattening layer converts the output of the previous layer to a one-dimensional vector , and the full connected layer takes that vector and calculates a final classification output.