**Convolutional Neural Networks**

**University of Luxembourg**

***Sean Achtatou***

The Convolutional Neural Networks, is a Neural Networks based on the use of the Convolution on inputs data, a.k.a convolutional layer. The convolutional Neural Networks is mainly used in computer vision for Machine and Deep Learning to recognize different types of objects, people on images or simply digit, but can be used for any other Deep Learning problem.

There exist different types of Convolutional Neural Networks with different architecture. For our project, we used *YOLO*[6-7], which is a Convolutional Neural Networks trained to be used for human, and others objects detection. Nevertheless, the Convolutional Neural Networks operations mainly all follow the same architecture which is the following.

The processing of the convolutional neural networks is based on multiples steps[4]:

1. **Convolution**
2. **Padding (optional)**
3. **Pooling**
4. **Stride**
5. **Filters (for convolution)**
6. **Activation Function**
7. **Fully connected neural networks**

We will now introduce each of the steps of the Convolutional Neural Networks. (you have the possibility to follow the steps along the PowerPoint “Convolutionial\_Neural\_Networks\_Steps.pptx”)

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

1. **Convolution**

We can define the convolution as the modification and transformation of an input data to another output. The result is obtained by a mathematical operation between two functions, resulting into a third one being the results.

The Convolutional layers are a set of 2D (or 3D) filters equally equals in both axis, but smaller than the original input. With an input image, we can use the filters to convolve with the input, producing a new output by summing the element-wise multiplication of the input with the filters. The use of the filters can be different, each filters has a different resulting pixels, some of filters will provide information about edge of the image while others will smooth the image. The numbers of filters depend on the amount of details we would like to obtain about the image, more filters mean more precision.

* 1. **Filters**

The filters are used through each convolution steps, there are used to produce an output providing different information about the input data. There exist different types of filters, some can provide information about edge detection, other can just be identity (not really useful). The filters can be of different size (but each axis must be of the same size) but must always be inferior to the input data with which it will convolve.

**Note:**

The convolutional layers work by using one or multiples filters on an image. The output of this process is a volume block containing each features of the input data. Assume, we have an image of 16x16 with 4 filters of 3x3, the output is a block 14x14x4.

* 1. **Stride**

The Stride is how the convolution is done with the filters. Indeed, the filter can slide through the input data one by one (called stride 1) or more. However, more the stride is high, more the output sized will be reduced and less information we will get. Still, the stride cannot go above half of the size of the input data. If the stride is different to one, we need as well to be careful to the size of the input data, indeed it must not go beyond the limit of the input data, to solve this problem we can use a padding.

* 1. **Padding**

With the convolution we used filters of 3x3 for an image of 4x4, the output was of 2x2. However, it could happen that we would like to have the same size of the output as the size of the input image to keep all the original information.

To do that, we can use padding, which will add around the original input image additional arrays of pixels of zero values, this is called same padding since it produces a same input and output size. Based on the actual filters size, we can add one or more padding to the input. The padding can as well be useful if we decide to change the stride of the filters, in this case, the use of one or more padding can help to provide a good convolution.

1. **Pooling**

In an image, most of the pixel’s neighbors to each other tens to have similar pixels’ values, meaning in the process of the convolution most of the pixels processing will produce same and useless results, nothing really new and useful to detection is found, moreover it will reduce the dimensionality size and accelerate the processing. To help solve this problem, and save computation time, we can use pooling layers, they simply reduce the size of the output by pooling pixels together. Pooling is mainly done by using max, minor average to a set of neighbor’s pixels. Today, the average pooling is the most used keeping the most important information. The pooling layers’ size can go from two to a reasonable number based on the image size. More the pooling layer size is high, less information we will obtain as output.

1. **Fully connected neural networks[1]**

After having convolve the input data multiples times (can be three to four times for example – convolve, pool) we will take the final output to input it the a fully connected neural networks. The input layer of the neural networks will be the flatten data of the final output data we would have obtain in the convolution. Then we would use the neural networks as we would for any other one.

1. **Activation function[9]**

We have not talked about the activation function, but there are very important for the convolutional neural networks to work. The activation function will help the convolution to introduce a non-linearity to the data obtained. One of the most known and performant activation function is the ReLU, it will mainly transform all the negative values to 0 and keep the others. However, we can have the possibility to use other different activation function based on the performance and the result we would like to obtain. In the Convolutional Neural Networks, the activation function ReLU is used just after the convolution and before the pooling.

To have a prediction of our Convolutional Neural Networks, we need an activation function for each layers in the fully connected neural networks as well. The activation function will provide how the different layers and the neural networks will predict the output. We can use ReLU for each hidden layers and Softmax for the output layer.

***Sources:***

[1]<http://neuralnetworksanddeeplearning.com/chap1.html#complete_zero>

[2]<https://ujjwalkarn.me/2016/08/11/intuitive-explanation-convnets/>

[3]<https://victorzhou.com/blog/intro-to-cnns-part-1/>

[4]<https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148>

[5]<https://www.youtube.com/watch?v=vT1JzLTH4G4&list=PL3FW7Lu3i5JvHM8ljYj-zLfQRF3EO8sYv&index=2&t=0s>

[6]<https://pjreddie.com/darknet/yolo/>

[7]<https://github.com/wizyoung/YOLOv3_TensorFlow>

[8]<https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0>

[9] <https://en.wikipedia.org/wiki/Activation_function>