

July 14

I. <https://poloclub.github.io/cnn-explainer/>

II. Response

- A. Last week you did an exercise where you manually applied a 3x3 array as a filter to an image of two people ascending an outdoor staircase. Modify the existing filter and if needed the associated weight in order to apply your new filters to the image 3 times. Plot each result, upload them to your response, and describe how each filter transformed the existing image as it convolved through the original array and reduced the object size. What are you functionally accomplishing as you apply the filter to your original array (see the following snippet for reference)? Why is the application of a convolving filter to an image useful for computer vision? Stretch goal: instead of using the `misc.ascent()` image from `scipy`, can you apply three filters and weights to your own selected image? Again describe the results.

```
#Convolution
for x in range(1,size_x-1):
    for y in range(1,size_y-1):
        convolution = 0.0
        convolution = convolution + (i[x - 1, y-1] * filter[0][0])
        convolution = convolution + (i[x, y-1] * filter[0][1])
        convolution = convolution + (i[x + 1, y-1] * filter[0][2])
        convolution = convolution + (i[x-1, y] * filter[1][0])
        convolution = convolution + (i[x, y] * filter[1][1])
        convolution = convolution + (i[x+1, y] * filter[1][2])
        convolution = convolution + (i[x-1, y+1] * filter[2][0])
        convolution = convolution + (i[x, y+1] * filter[2][1])
        convolution = convolution + (i[x+1, y+1] * filter[2][2])
        convolution = convolution * weight
        if(convolution<0):
            convolution=0
        if(convolution>255):
            convolution=255
        i_transformed[x, y] = convolution
```

- B. Another useful method is pooling. Apply a 2x2 filter to one of your convolved images, and plot the result. In effect what have you accomplished by applying this filter? Can you determine from the code which type of pooling filter is applied, and the method for selecting a pixel value (see the following snippet)? Did the result increase in size or decrease? Why would this method be method? Stretch goal: again, instead of using `misc.ascent()`, apply the pooling filter to one of your transformed images.

```
# Pooling

new_x = int(size_x/2)
new_y = int(size_y/2)
newImage = np.zeros((new_x, new_y))
for x in range(0, size_x, 2):
    for y in range(0, size_y, 2):
        pixels = []
        pixels.append(i_transformed[x, y])
        pixels.append(i_transformed[x+1, y])
        pixels.append(i_transformed[x, y+1])
        pixels.append(i_transformed[x+1, y+1])
        pixels.sort(reverse=True)
        newImage[int(x/2),int(y/2)] = pixels[0]
```

- C. The lecture for today (Coding with Convolutional Neural Network) compared the application of our previously specified deep neural network with a newly specified convolutional neural network. Instead of using the fashion_MNIST dataset, use the mnist dataset (the hand written letters) to train and compare your DNN and CNN output. Were you able to improve your model by adding the Conv2D and MaxPooling2D layers to your neural network? Plot the convolutions graphically, include them in your response and describe them. Edit the convolutions by changing the 32s to either 16 or 64 and describe what impact this had on accuracy and training time. What happens if you add more convolution layers?
- D. For tomorrow
1. Lecture 5 from ML Foundations - Classifying real-world images
 2. Loading and preprocessing images and text data
 3. Daily exercises
 4. Thursday: Project 2