ASSIGNMENT#1

1.What are Channels and Kernels (according to EVA)?

Kernel: A kernel is defined in a convolutional layer for extracting the information, compute and inform neuron about the feature. Kernels are convolved over images and extracts specific features. Kernels are more of like a 3x3 matrix which are also known as filters and feature extractors.

Channel: Channels are collection of extracted feature sets which will be passed to next layers. An image can have any number of channels. The appropriate number is better to be decided by AI.

2. Why should we (nearly) always use 3x3 kernels?

- The smaller kernels like 1x1, are mainly used for dimensionality reduction. These kernels would capture just one pixel of the feature map. So, the features extracted will be finely grained and will not consists of any neighboring pixels' information.
- The kernels like 2x2, 4x4…etc (even-sized matrices) are not used as it creates the distortions across the layer as it breaks the symmetry. The larger sized kernels would take longer time for training and cost factor makes larger kernels of least choice.
- On the other hand, 3x3 is an odd-sized kernel which can be symmetric in nature and same receptive field can be achieved using multiple layers of 3x3 kernels with smaller number of parameters, instead of larger sized kernels like 5x5, 7x7…etc. The hardware manufacturers have optimized the hardware required for processing the model. Therefore, 3x3 kernels are optimal choice.

3. How many times do we need to perform 3x3 convolutions operations to reach close to 1x1 from 199x199 (type each layer output like 199x199 > 197x197...)

The number of convolution operations required to reach 1x1 from 199x199 is **"99"** and computation is as follows:

199x199 | 3x3 = 197x197, 197x197 | 3x3 = 195x195, 195x195 | 3x3 = 193x193, 193x193 | 3x3 = 191x191, 191x191 | 3x3 = 189x189, 189x189 | 3x3 = 187x187, 187x187 | 3x3 = 185x185, 185x185 | 3x3 = 183x183, 183x183 | 3x3 = 181x181, 181x181 | 3x3 = 179x179, 179x179 | 3x3 = 177x177, 177x177 | 3x3 = 175x175, 175x175 | 3x3 = 173x173, 173x173 | 3x3 = 171x171, 171x171 | 3x3 = 169x169, 169x169 | 3x3 = 167x167, 167x167 | 3x3 = 165x165, 165x165 | 3x3 = 163x163, 163x163 | 3x3 = 161x161, 161x161 | 3x3 = 159x159, 159x159 | 3x3 = 157x157, 157x157 | 3x3 = 155x155, 155x155 | 3x3 = 153x153, 153x153 | 3x3 = 151x151, 151x151 | 3x3 = 149x149, 149x149 | 3x3 = 147x147, 147x147 | 3x3 = 145x145, 145x145 | 3x3 = 143x143, 143x143 | 3x3 = 141x141, 141x141 | 3x3 = 139x139, 139x139 | 3x3 = 137x137, 137x137 | 3x3 = 135x135, 135x135 | 3x3 = 133x133, 133x133 | 3x3 = 131x131, 131x131 | 3x3 = 129x129, 129x129 | 3x3 = 127x127, 127x127 | 3x3 = 125x125, 125x125 | 3x3 = 123x123, 123x123 | 3x3 = 121x121, 121x121| 3x3 = 119x119, 119x119 | 3x3 = 117x117, 117x117 | 3x3 = 115x115, 115x115 | 3x3 = 113x113, 113x113 | 3x3 = 111x111, 111x111 | 3x3 = 109x109, 109x109 | 3x3 = 107x107, 107x107 | 3x3 = 105x105, 105x105 | 3x3 = 103x103, 103x103 | 3x3 = 101x101, 101x101 | 3x3 = 99x99, 99x99 | 3x3 = 97x97, 97x97 | 3x3 = 95x95, 95x95 | 3x3 = 93x93, 93x93 | 3x3 = 91x91, 91x91 | 3x3 = 89x89, 89x89 | 3x3 = 87x87, 87x87 | 3x3 = 85x85, 85x85 | 3x3 = 83x83, 83x83 | 3x3 = 81x81, 81x81 | 3x3 = 79x79, 79x79 | 3x3 = 77x77, 77x77 | 3x3 = 75x75, 75x75 | 3x3 = 73x73, 73x73 | 3x3 = 71x71, 71x71 | 3x3 = 69x69, 69x69 | 3x3 = 67x67, 67x67 | 3x3 = 65x65, 65x65

| 3x3 = 63x63, 63x63 | 3x3 = 61x61, 61x61 | 3x3 = 59x59, 59x59 | 3x3 = 57x57, 57x57 | 3x3 = 55x55, 55x55 | 3x3 = 53x53, 53x53 | 3x3 = 51x51, 51x51 | 3x3 = 49x49, 49x49 | 3x3 = 47x47, 47x47 | 3x3 = 45x45, 45x45 | 3x3 = 43x43, 43x43 | 3x3 = 41x41, 41x41 |3x3 = 39x39, 39x39 | 3x3 = 37x37, 37x37 | 3x3 = 35x35, 35x35 | 3x3 = 33x33, 33x33 | 3x3 = 31x31, 31x31 | 3x3 = 29x29, 29x29 | 3x3 = 27x27, 27x27 | 3x3 = 25x25, 25x25 | 3x3 = 23x23, 23x23 | 3x3 = 21x21, 21x21 | 3x3 = 19x19, 19x19 | 3x3 = 17x17, 17x17 | 3x3 = 15x15, 15x15 | 3x3 = 13x13, 13x13 | 3x3 = 11x11, 11x11 | 3x3 = 9x9, 9x9 | 3x3 = 7x7, 7x7 | 3x3 = 5x5, 5x5 | 3x3 = 3x3, 3x3 | 3x3 = 1.

4. How are kernels initialized?

The kernel initialization is an important design step when developing the deep neural network and this initialization is random in nature which in general is done using pseudo random generator or Gaussian distribution technique with mean 0, standard deviation as 1.

5. What happens during the training of a DNN?

Learning will tune the kernel weights from random values to fine values with multiple feed-forward, loss calculation and back propagation, this process is called epoch. After multiple epochs, the kernel weights, which gives less error in prediction can be used for evaluation.