

ML Problem Solving

Discrete Convolutions

Given is a single 4x4 image I and a 2x2 kernel K , both stored as 2D arrays. For centering the kernel w.r.t. the image, consider the top-left pixel of the kernel to be its center. The discrete convolution is given by $C(x, y, m, n) = \sum_M \sum_n I(x+m, y+n) \cdot K(m, n)$.

$I = \begin{bmatrix} 255 & 128 & 128 & 0 \\ 51 & 255 & 0 & 128 \\ 51 & 128 & 128 & 0 \\ 0 & 128 & 0 & 0 \end{bmatrix}$, $K = \begin{bmatrix} 1 & 2 \\ 0 & 2 \end{bmatrix}$

1. Normalize the pixels of I to the $[0, 1]$ range.

2. Calculate the output array of the 2-D discrete convolution of I and K , assuming same padding and stride of 1.

3. Calculate the output array of the 2-D discrete convolution of I and K , assuming valid padding and stride of 1.

4. Calculate the output array of the 2-D discrete convolution of I and K , assuming valid padding and stride of 2.
5. Given the following NN architecture, and the original shape of $x = [4, 1, 28, 28]$, what is the size of the output tensor for each layer? Layers:
- (1) `nn.Conv2d(in_channels=1, out_channels=16, kernel_size=3, padding=valid, stride=1)`
 - (2) `nn.ReLU()`
 - (3) `nn.Conv2d(in_channels=16, out_channels=64, kernel_size=3, padding=valid, stride=2)`
 - (4) `nn.ReLU()`
 - (5) `nn.MaxPool2d(kernel_size=2)`
 - (6) `nn.Dropout(p=0.5)`