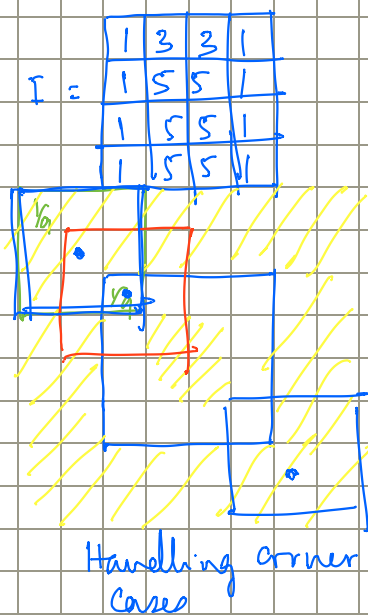
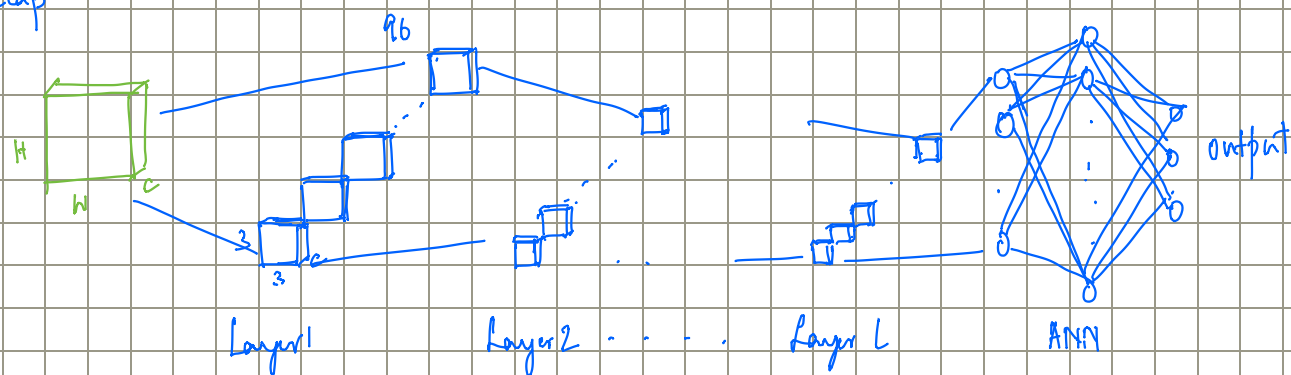


29/9/23

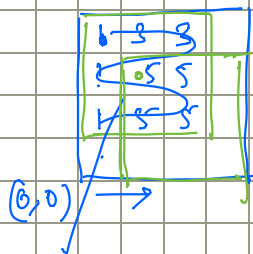
Deep Learning

- Recap: CNN, AlexNet, VGG
- $f_{\text{CNN}}(x; \theta)$
- Training a CNN
- Recap



$$h = \begin{bmatrix} x_0 & x_1 & x_2 \\ x_3 & x_4 & x_5 \\ x_6 & x_7 & x_8 \end{bmatrix}$$

$$J = I \times h$$



$$J(1,2) = \frac{1}{9} \cdot 1 + \frac{1}{9} \cdot 3 + \frac{1}{9} \cdot 3 + \frac{1}{9} \cdot 1 + \frac{1}{9} \cdot 5 + \frac{1}{9} \cdot 5 + \frac{1}{9} \cdot 1 + \frac{1}{9} \cdot 5 + \frac{1}{9} \cdot 5$$

$$= \frac{1}{9} [1 + 3 + 3 + 1 + 5 + 5 + 1 + 5 + 5]$$

$$= \frac{29}{9} \approx 3.2$$

$$J(2,1) = \frac{1}{9} (5 \times 6 + 3 \times 1)$$

$$J(0,2) = \frac{1}{9} (5 \times 0 + 2 \times 1 + 3 + 5)$$

The feature map finally is $K = \sigma(I)$

Recall: $\underbrace{f_{\text{ANN}}(z; \theta)} = g(\langle \underline{z}, \underline{\theta} \rangle)$

$$f_{\text{CNN}}(x; \theta) = f_{\text{ANN}} \circ f_L \circ f_{L-1} \dots \circ \underbrace{f_1}_{\text{first conv layer}}$$

$$f_1^i = \sigma(h_1^i * I)$$

$$f_l^c = \sigma(\underline{h}_x^l * \underline{f}_{l+1})$$

- Typically, CNNs have a pooling layer at the output of each conv. layer.
- Striding:

