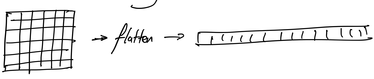
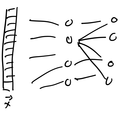
Convolutional Layers and Images

Wednesday, July 19, 2023 2:07 PM

How can we handle images?



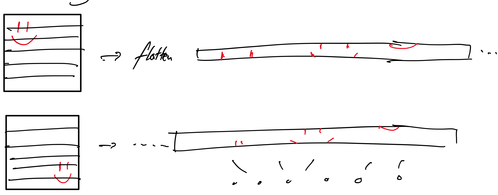


for a 10-class classification problem, what is the size of the output?

10 - one for each class

if using softmax

Problems with images as vectors

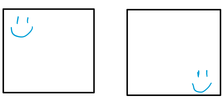


- different neurons have to deal with the same features, depending on where they are in the image
- redundant effort - very inefficient

Convolution

3×3 "filter"

- the filter slides across the whole image, "looking" at all of it
- recall: $w \cdot x$ in a neuron is a measure of similarity - how much does this filter "recognize" what it sees?

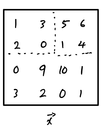


- "smile detector" activates regardless of the location of the smile in the image
- translational invariance

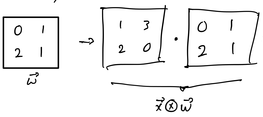
Q: is the smile detector going to activate?

translational invariance ≠ rotational invariance

Grayscale Example



$\vec{x}$

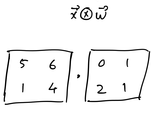


$\vec{w}$

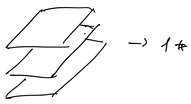
$\vec{x} \circledast \vec{w}$

$1\times0 + 3\times1 + 2\times2 + 0\times1 = 0 + 3 + 4 + 0 = 7$

stride-2



2×2 output



→ 1 filter

stride-1 → 3×3

$\vec{x}$ (6×8)

$\vec{w}_0$ (2×2)

$\vec{w}_{F-1}$

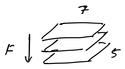
$\rightarrow \sigma(\vec{x} \circledast \vec{w}_0 + b_0)$

stride-1 (5,7)

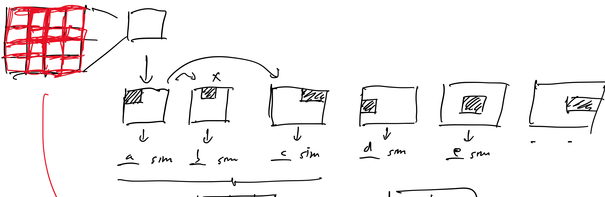
$\rightarrow \sigma(\vec{x} \circledast \vec{w}_1 + b_1)$

output: (F, 5, 7)

# of filters



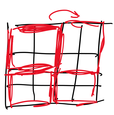
(sheet, rows, cols)



a sim b sim c sim d sim e sim

stride-1

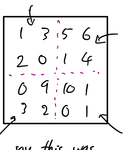
stride-2



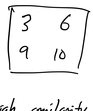
2×2

Max Pool

- operation that often follows one or more conv layers
  - purely a discretization/downsampling process - no learnable parameters
  - $f(\vec{x}) = \max(\vec{x})$



→ 2×2 max pool w/ stride 2 →



say this was the result of $\vec{w} \circledast \vec{x}$

high values come from high similarity

∴ max pool helps later parts of the network "focus on" high-similarity detections and reject low-similarity and redundant detections

"translation invariance"