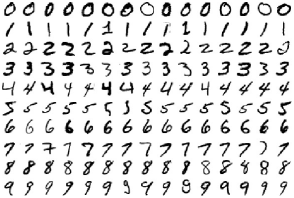
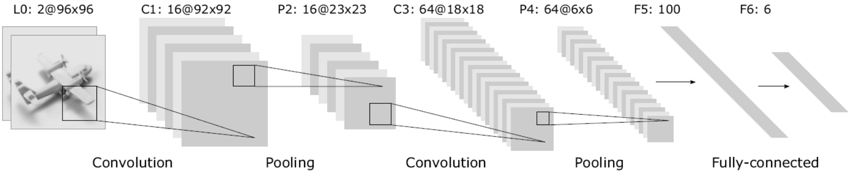
Image Processing Basics

COMP4211

MNIST: 10 classes (digits 0to 9)



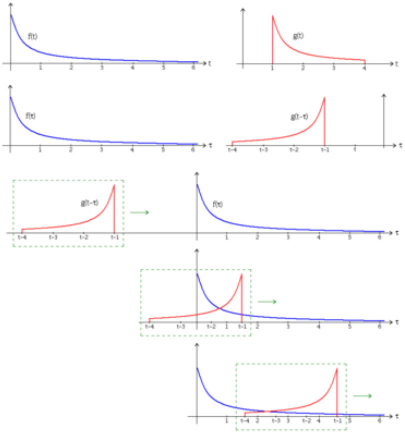
Convolutional neural network



Continuous Convolution

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\,d\tau$$

- demo



- commutative: $f * g = g * f$
- associative: $f * (g * h) = (f * g) * h$

Discrete Convolution

$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\,d\tau$$

Discrete convolution:

$$\boxed{(f * g)(m) = \sum_{n=0}^{N-1} f(n)g(m-n)}$$

2D Convolution

image $I(i,j)$; kernel (mask) K

- convolution

$$S(i,j) = (I * K)(i,j) = \sum_m \sum_n I(m,n) K(i-m, j-n)$$

convolution is commutative

$$S(i,j) = (K * I)(i,j) = \sum_m \sum_n I(i-m, j-n) K(m,n)$$

- easier to implement

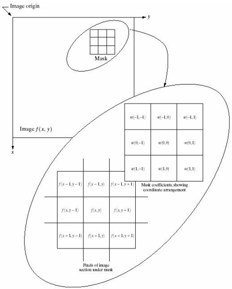


FIGURE 3.32 The mechanics of spatial filtering. The magnified drawing shows a 3×3 mask and the image section directly under it; the image section is shown displaced out from under the mask for ease of readability.

we have flipped the kernel relative to the input

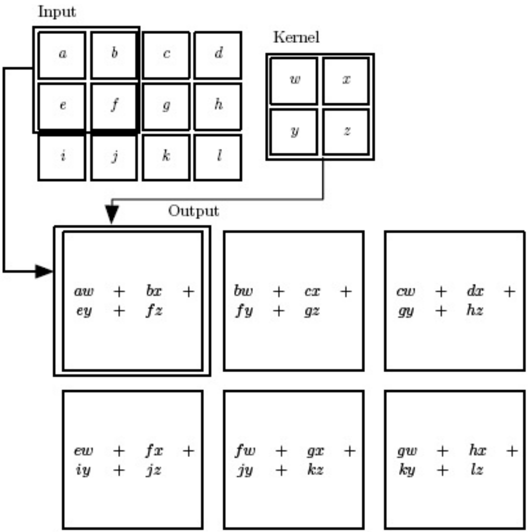
- good for writing proofs, not an important property of a neural network implementation

more common approach

- same as convolution but without flipping the kernel

$$S(i,j) = (K * I)(i,j) = \sum_m \sum_n I(i+m, j+n) K(m,n)$$

2D Convolution...



Smoothing (Averaging) Filter

$$1/9 \; * \; \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

- window size

original

n=5 (nxn mask)

n=15 (nxn mask)

n=25 (nxn mask)

Other Arrangements

1	1	1
1	2	1
1	1	1

1	1	1	1	1
1	2	3	2	1
1	3	4	3	1
1	2	3	2	1
1	1	1	1	1

center pixel: 1 vs 5



Sharpening Filters

Averaging pixels

- blur
- analogous to integration, related to sum of pixel intensity values

Differentiation

- has the opposite effect of blurring
- sharpens an image, related to difference between intensity values

First derivative

$$\frac{\partial f}{\partial x} \leftrightarrow f(x+1) - f(x)$$

Edge Detector

-1	0	1
-1	0	1
-1	0	1

-1	-1	-1
0	0	0
1	1	1