J										
		3.6								

$$J(1) = I(0) \times h(-1) + I(1) \times h(0) + I(2) \times h(1)$$

$$J(1) = 5 \times \frac{1}{3} + 4 \times \frac{1}{3} + 2 \times \frac{1}{3}$$

	0	1	2	3	4 . . .					
I	5	4	2	3	7	4	6	5	3	6

	-1	0	1
h	1/3	1/3	1/3

$$J(x) = \sum_{i=-1}^{1} h(i) \times I(x+i)$$

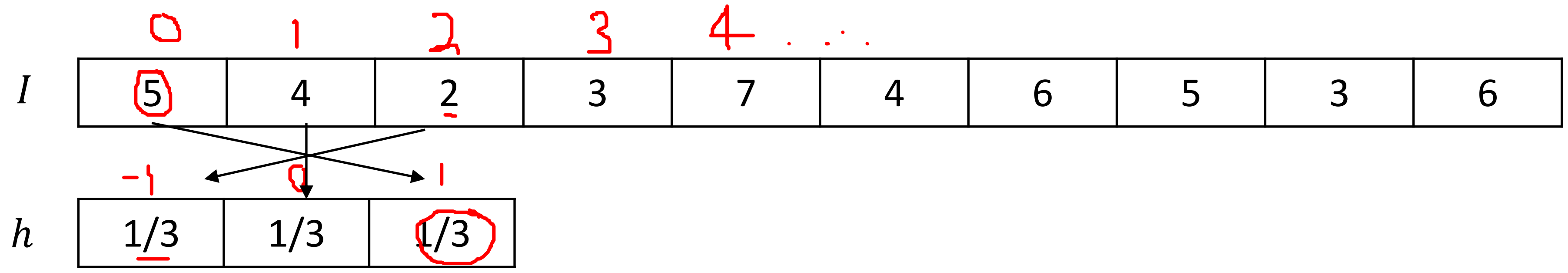
$$J = I \circ h = \sum_{i=-N}^{N} h(i) \times I(x+i)$$

$$J = I \circ h = \sum_{i=-N}^{N} \sum_{j=-N}^{N} h(i,j) \times I(x+i,,y+j)$$

Convolution

Convolution is just like correlation, except that we flip over the filter before correlating.

Correlation and convolution are identical when the filter is symmetric.



$$J(x) = \sum_{i=-1}^{1} h(i) \times I(x-i)$$

$$J = I * h = \sum_{i=-N}^{N} h(i) \times I(x-i)$$

$$J = I * h = \sum_{i=-N}^{N} \sum_{j=-N}^{N} h(i,j) \times I(x-i,,y-j)$$

The key difference between convolution and correlation is that convolution is associative.

$$F * (G * I) = (F * G) * I$$

Kernel Size: The kernel size defines the field of view of the convolution. A common choice for 2D is 3 — that is 3x3 pixels.

Padding: The padding defines how the border of a sample is handled.

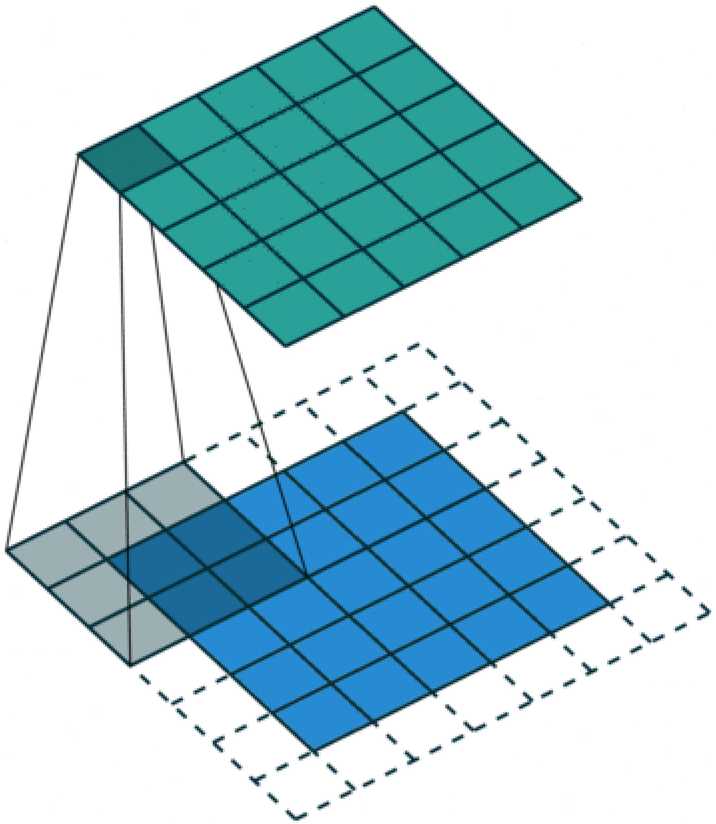


Image Size after convolution

Image size : n x n , filter size : f x f

Image size after convolution : $(n-f+1) \times (n-f+1)$

With zero padding: $(n+2p-f+1) \times (n+2p-f+1)$

Separable Filters

Generally, 2D convolution is more expensive than 1D convolution because the sizes of the filters we use are larger.

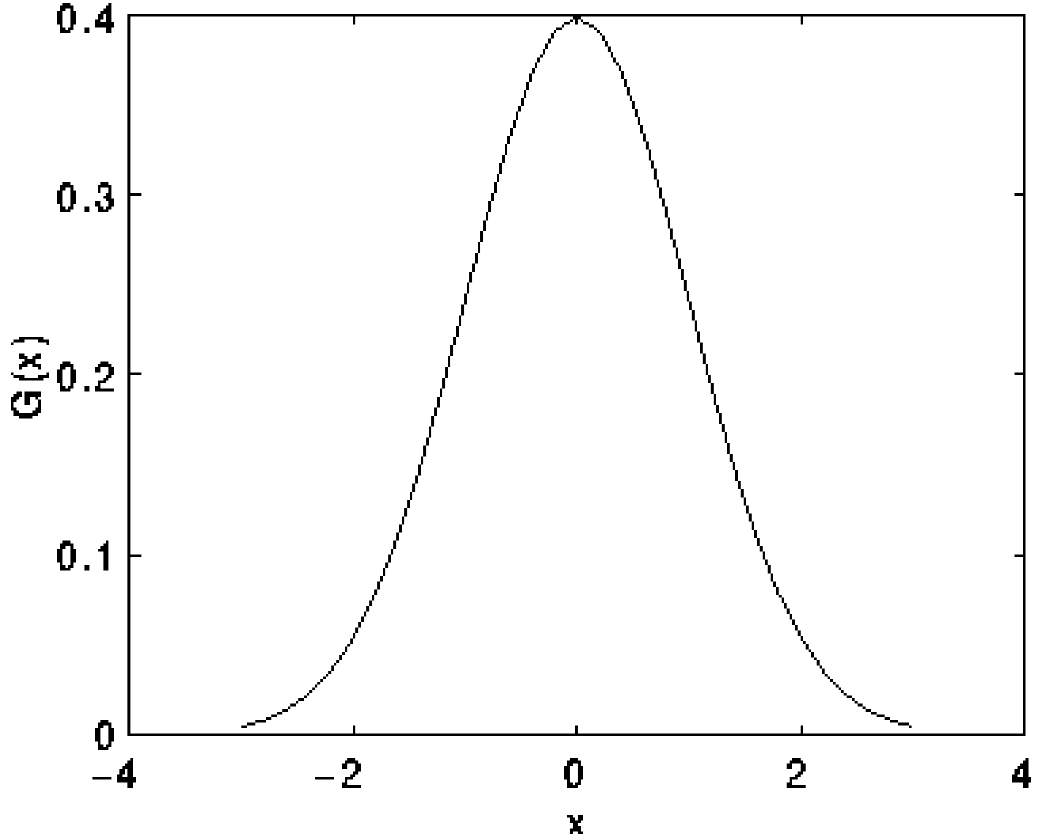
If our filter is N x N in size, and our image contains M x M pixels, then the total number of multiplications we must perform is N^2M^2.

With an important class of filters, we can save on this computation.

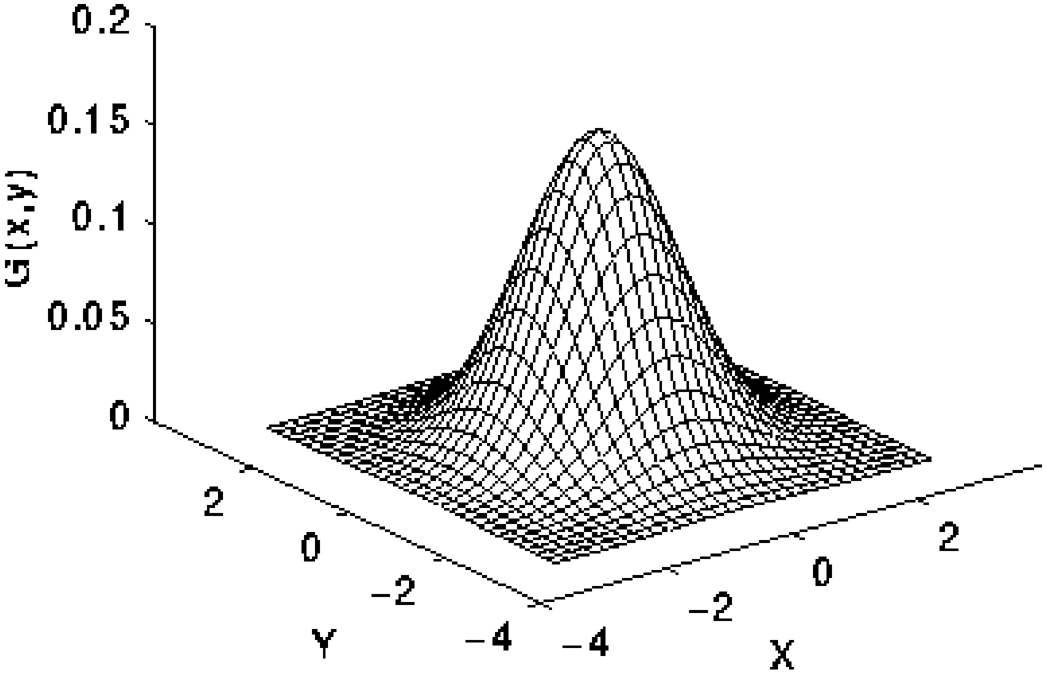
$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{3} & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & \frac{1}{3} & 0 \end{bmatrix} \circ \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & 0 \end{bmatrix}$$

Gaussian Smoothing

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$



$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$



$\frac{1}{273}$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

2-D Gaussian distribution with mean (0,0) and =1

Properties of Convolution

- Commutativity

Same result if we convolve
image with filter or vice versa

$$I * H = H * I$$

- Linearity

If image multiplied by scalar
Result multiplied by same scalar

$$(s \cdot I) * H \quad = \quad I * (s \cdot H) \quad = \quad s \cdot (I * H)$$

$$(I_1 + I_2) * H \quad = \quad (I_1 * H) + (I_2 * H)$$

If 2 images added and convolve
result with a kernel *H*,
Same result if each image
is convolved individually + added

(notice) $$(b + I) * H \quad \neq \quad b + (I * H)$$

- Associativity

$$A * (B * C) = (A * B) * C$$

Order of filter application irrelevant
Any order, same result

Properties of Convolution

- Separability

$$H = H_1 * H_2 * \ldots * H_n$$

$$\begin{aligned} I * H &= I * (H_1 * H_2 * \ldots * H_n) \\ &= (\ldots((I * H_1) * H_2) * \ldots * H_n) \end{aligned}$$

- If a kernel H can be separated into multiple smaller kernels

Applying smaller kernels H_1 H_2 … H_N H one by one computationally cheaper than apply 1 large kernel H

$$H = H_1 * H_2 * \ldots * H_n$$

Computationally More expensive

Computationally Cheaper

Gaussian Kernel

- 1D

$$g_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right)$$

- 2D

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

Separability of 2D Gaussian

- 2D gaussian is just product of 1D gaussians:

$$
\begin{aligned}
G_\sigma(x, y) &= \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \\
&= \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right) \cdot \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{y^2}{2\sigma^2}\right) \\
&= g_\sigma(x) \cdot g_\sigma(y)
\end{aligned}
$$

Separable!

Separability of 2D Gaussian

- Consequently, convolution with a gaussian is separable

$$I * G = I * G_x * G_y.$$

- Where G is the 2D discrete gaussian kernel;
- G_x is "horizontal" and G_y is "vertical" 1D discrete Gaussian kernels