Digital filters

A filter is usually a local transform.

For an 1D signal it is

$$f(x) \to f'(x) = \int_S f(x - x')g(x')\,dx',$$

and for an 2D image

$$f(x, y) \to f'(x, y) = \int_S f(x - x', y - y')g(x', y')\,dx'\,dy',$$

where S is a small neighbourhood of the origin. Many filters have the form of a convolution; it is often possible to work either with the original image or its Fourier transform.

Nonrecursive filter: all new values are evaluated using the original signal or image only.

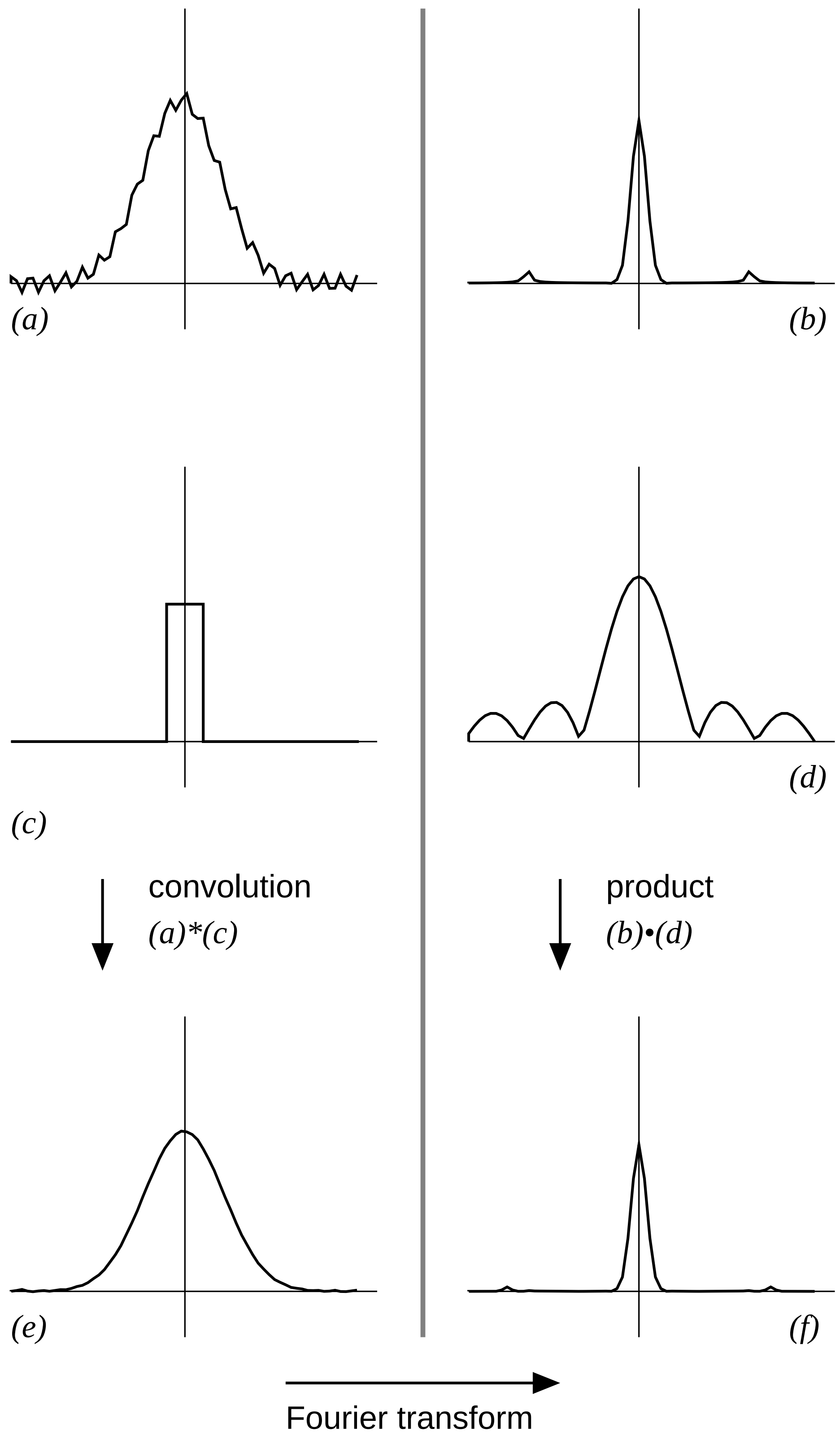
Recursive filter: each pixel is replaced by a new value as soon as it has been computed. The transformed signal/image can be stored in the space of the original one.

Low-pass filter: remove all frequencies higher than a given limit. Most easily done using the Fourier transform. The filter has a smoothing effect.

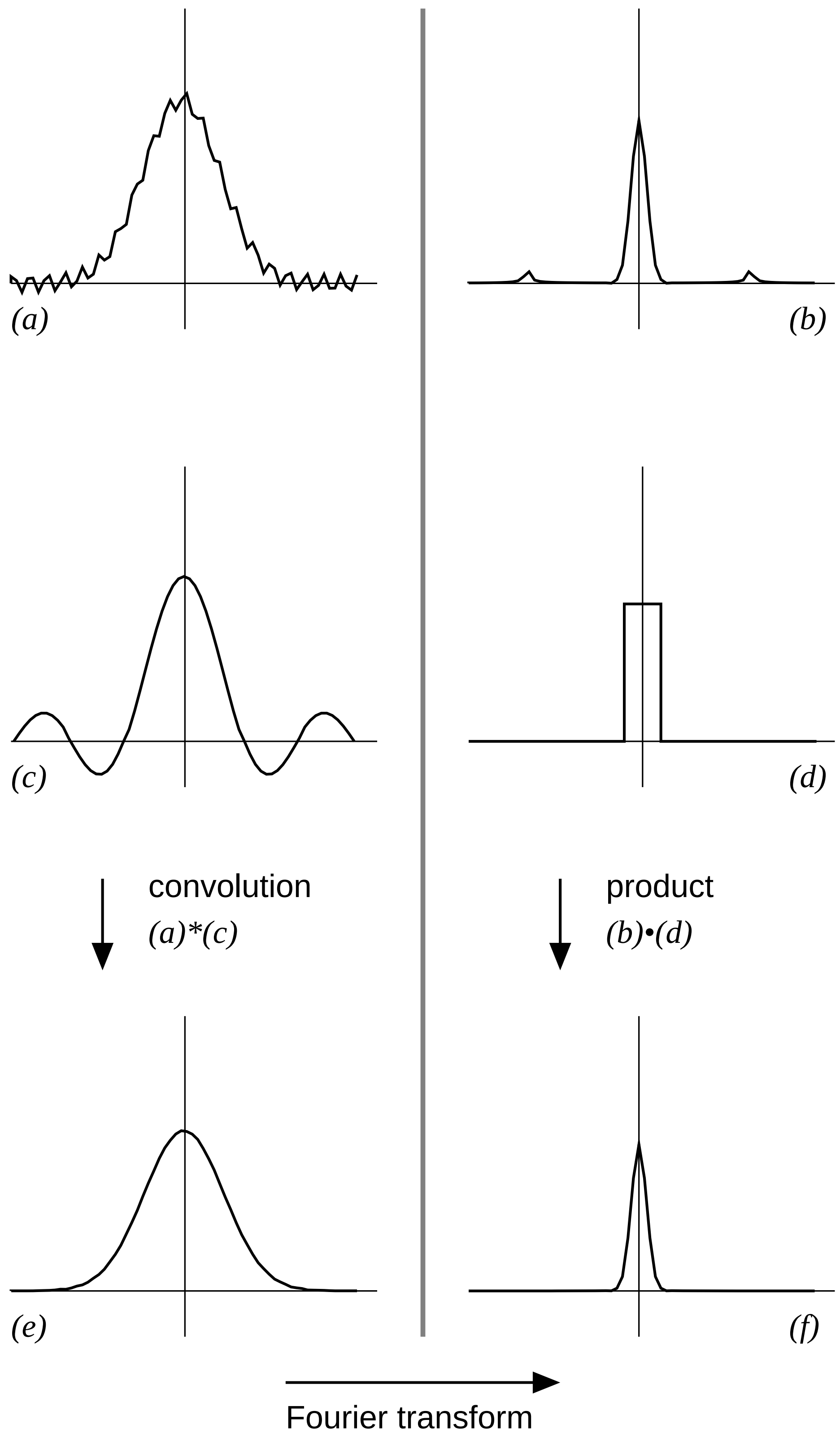
The Fourier transform of the rect function (a rectangular pulse) is a sinc function, whose nonzero values are not limited to a finite region.

Thus smoothing the original image by convolving with the rect function does not completely remove high frequencies.

High-pass filter: remove low frequencies, which has a sharpening effect. Corresponds to differentiation.



(a)
(b)
(c)
(d)
convolution
(a)*(c)
product
(b)•(d)
(e)
(f)
Fourier transform



(a)
(b)
(c)
(d)
convolution
(a)*(c)
product
(b)•(d)
(e)
(f)
Fourier transform

Low-pass filters (not quite):

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \frac{1}{5}\begin{bmatrix} & 1 & \\ 1 & 1 & 1 \\ & 1 & \end{bmatrix} \qquad \frac{1}{6}\begin{bmatrix} & 1 & \\ 1 & 2 & 1 \\ & 1 & \end{bmatrix}$$

E.g. the Laplace operator is a high-pass filter

$$\begin{aligned} &\nabla^2 f(i,j) = \Delta_x^2 f(i,j) + \Delta_y^2 f(i,j) \\ &= (f(i+1,j) + f(i-1,j) + f(i,j+1) + f(i,j-1)) - 4f(i,j). \end{aligned}$$

or

$$\begin{bmatrix} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{bmatrix}$$

Assume the image has been blurred by adding a fraction ϵ of the sum of neighbouring pixels to each pixel:

$$p(i,j) = f(i,j) + \epsilon n(i,j),$$

where

$$n(i,j) = f(i+1,j) + f(i-1,j) + f(i,j+1) + f(i,j-1).$$

Subtract an expression proportional to the Laplace operator:

$$\begin{aligned} & p(i,j) - k\nabla^2 p(i,j) \\ & = (1+4k)p - k(p(i+1,j) + p(i-1,j) + p(i,j+1) + p(i,j-1)) \\ & = (1+4k)f(i,j) + (1+4k)\epsilon n(i,j) - kn(i,j) \\ & \quad - k\epsilon(n(i+1,j) + n(i-1,j) + n(i,j+1) + n(i,j-1)). \end{aligned}$$

If k and ϵ are small, the last term can be omitted. If $k = \epsilon/(1-4\epsilon)$, the other two terms vanish, and we get:

$$p(i,j) - k\nabla^2 p(i,j) = (1+4k)f(i,j).$$

Thus the original image has been restored (within a constant factor).

Median filter: Averaging reduces sharp spikes (like cosmic rays), but does not remove them; it only spreads their energy to a wider are. A better solution is to replace each pixel by the median of a small neighbourhood. This does not blur edges or affect areas where the gradient is monotonous.