Filtering vs Convolution

16-385 Computer Vision (Kris Kitani)
Carnegie Mellon University

Filters we have learned so far …

The 'Box' filter

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Gaussian filter

$$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Sobel filter

$$\frac{1}{8}\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

Laplace filter

$$\frac{1}{8}\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Filtering vs Convolution

filtering (cross-correlation)

$$h = g \otimes f$$

output, filter, image

$$h[m,n] = \sum_{k,l} g[k,l] f[m+k, n+l]$$

What's the difference?

convolution

$$h = g \star f$$

$$h[m,n] = \sum_{k,j} g[k,l] f[m-k, n-l]$$

Credit: Steve Seitz

Filtering vs Convolution

filtering
(cross-correlation)

$$h = g \otimes f$$

$$\underset{\text{output}}{h[m,n]} = \sum_{k,l} \underset{\text{filter}}{g[k,l]}\, \underset{\text{image}}{f[m+k, n+l]}$$

filter flipped vertically
and horizontally

convolution

$$h = g \star f$$

$$h[m,n] = \sum_{k,j} g[k,l]\, f[m-k, n-l]$$

Filtering vs Convolution

output filter image

filtering
(cross-correlation)

$$h = g \otimes f \qquad h[m,n] = \sum_{k,l} g[k,l]\, f[m+k, n+l]$$

filter flipped vertically
and horizontally

convolution

$$h = g \star f \qquad h[m,n] = \sum_{k,j} g[k,l]\, f[m-k, n-l]$$

Suppose g is a Gaussian filter.
How does convolution differ from filtering?

Recall...

$$\frac{1}{16}\begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline\end{array}$$

Commutative

"can move stuff around"

$$a \star b = b \star a \ .$$

Associative

"can regroup things"

$$(((a \star b_1) \star b_2) \star b_3) = a \star (b_1 \star b_2 \star b_3)$$

Distributes over addition

"can take things through parenthesis"

$$a \star (b + c) = (a \star b) + (a \star c)$$

Scalars factor out

$$\lambda a \star b = a \star \lambda b = \lambda (a \star b)$$

Derivative Theorem of Convolution $\frac{\partial}{\partial x}(h \star f) = (\frac{\partial}{\partial x} h) \star f$

can precompute this

Derivative Theorem of Convolution

$$\frac{\partial}{\partial x}(h \star f) = (\frac{\partial}{\partial x}h) \star f$$

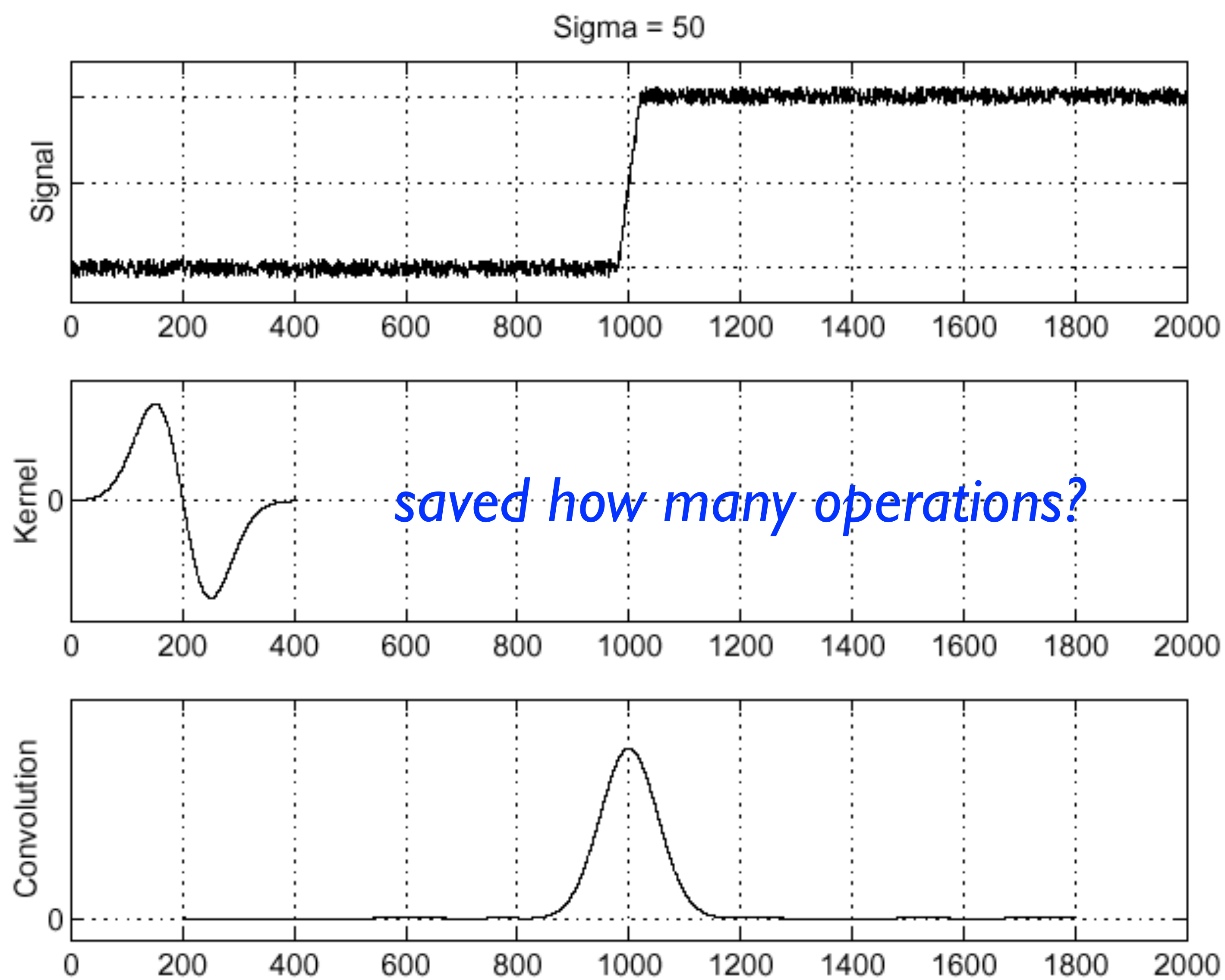
Input

Derivative of Gaussian

$\frac{\partial}{\partial x}h$

Output

$(\frac{\partial}{\partial x}h) \star f$



Recall ...

Input

Gaussian

Smoothed input

Derivative

Output