Image Filtering

Chetan Arora

Convolution

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Discrete Convolution

$$h(i) = (f * g)(i) = \sum_{k=-\infty}^{k=\infty} f(k)g(i-k)$$

$f(k)$

1	2	2	2	1

$g(k)$

1	0	-1

$g(-k)$

-1	0	1

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Discrete Convolution

$$h(i) = (f * g)(i) = \sum_{k=-\infty}^{k=\infty} f(k)g(i-k)$$

$f(k)$

0	0	0	1	2	2	2	1	0	0	0

$g(k)$

1	0	-1

$g(-k)$

-1	0	1

$g(0-k)$

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

0	0	0	1							

$h(0)$

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Discrete Convolution

$$h(i) = (f * g)(i) = \sum_{k=-\infty}^{k=\infty} f(k)g(i-k)$$

$f(k)$

0	0	0	1	2	2	2	1	0	0	0

$g(1-k)$

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

0	0	0	1	2						

$h(1)$

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Discrete Convolution

$$h(i) = (f * g)(i) = \sum_{k=1}^{n} f(k)g(i-k)$$

$f(k)$

0	0	0	1	2	2	2	1	0	0	0

$g(2-k)$

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

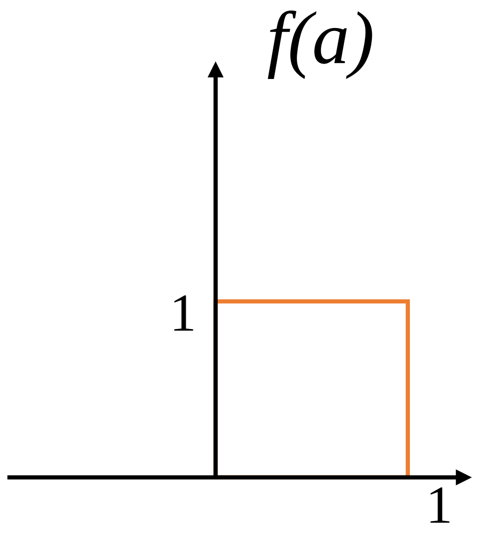
0	0	0	1	2	1					

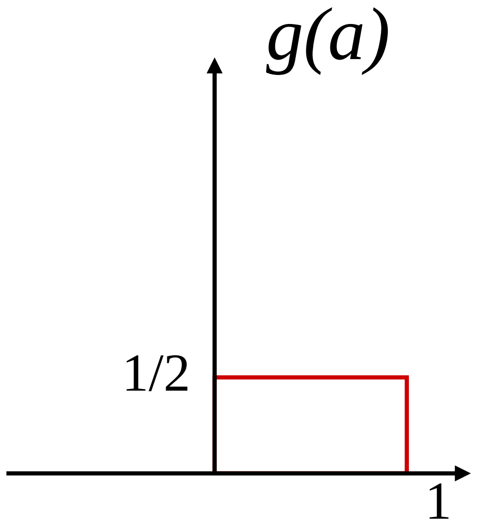
$h(2)$

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Continuous Convolution

$$(f * g)(x) = \int_{-\infty}^{\infty} f(a)g(x-a)da$$

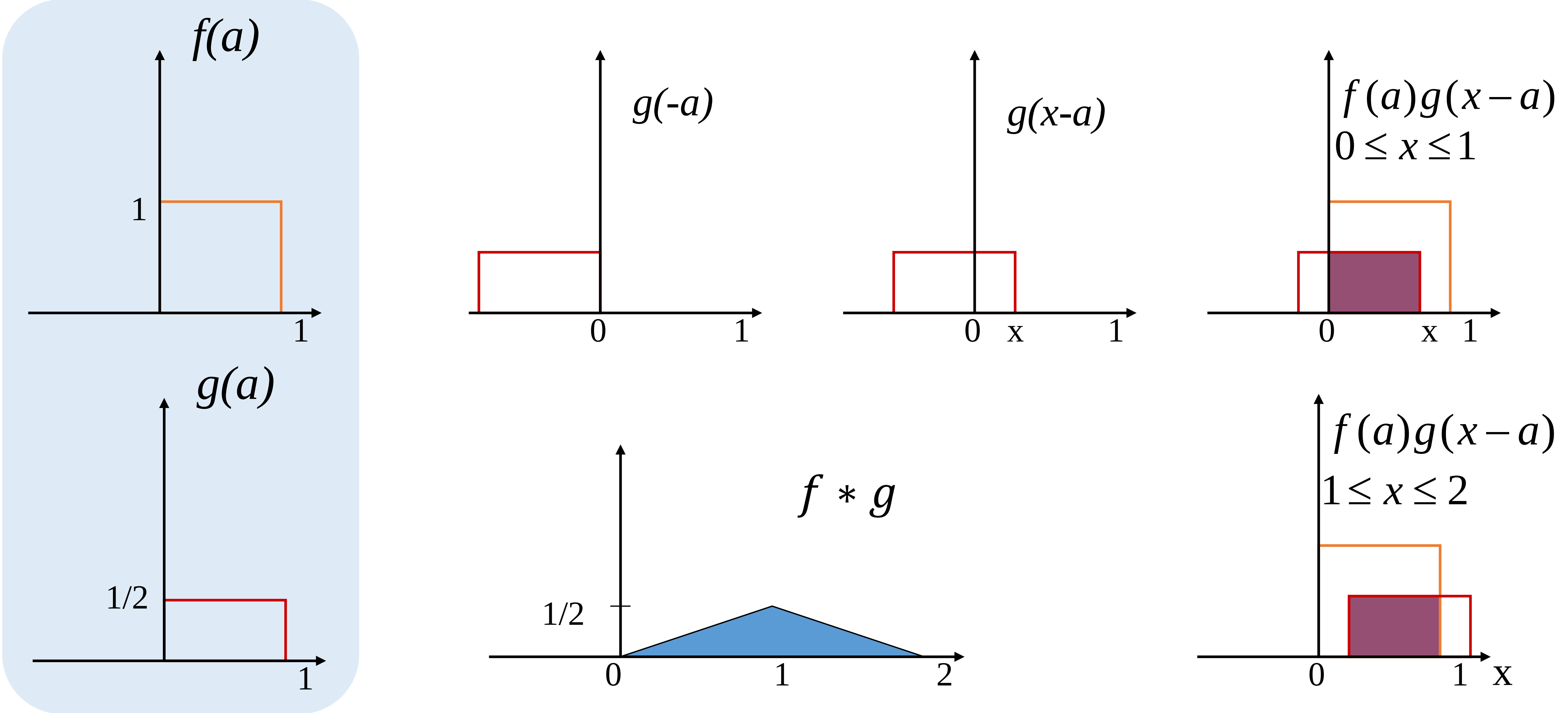




Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Continuous Convolution

$$(f * g)(x) = \int_{-\infty}^{\infty} f(a)g(x-a)da$$



Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Convolution

- **Why convolution?**
  - Response to Physics (sensors): Frequency response to color by sensors (color space - wavelength). Area response of sensors (image space)

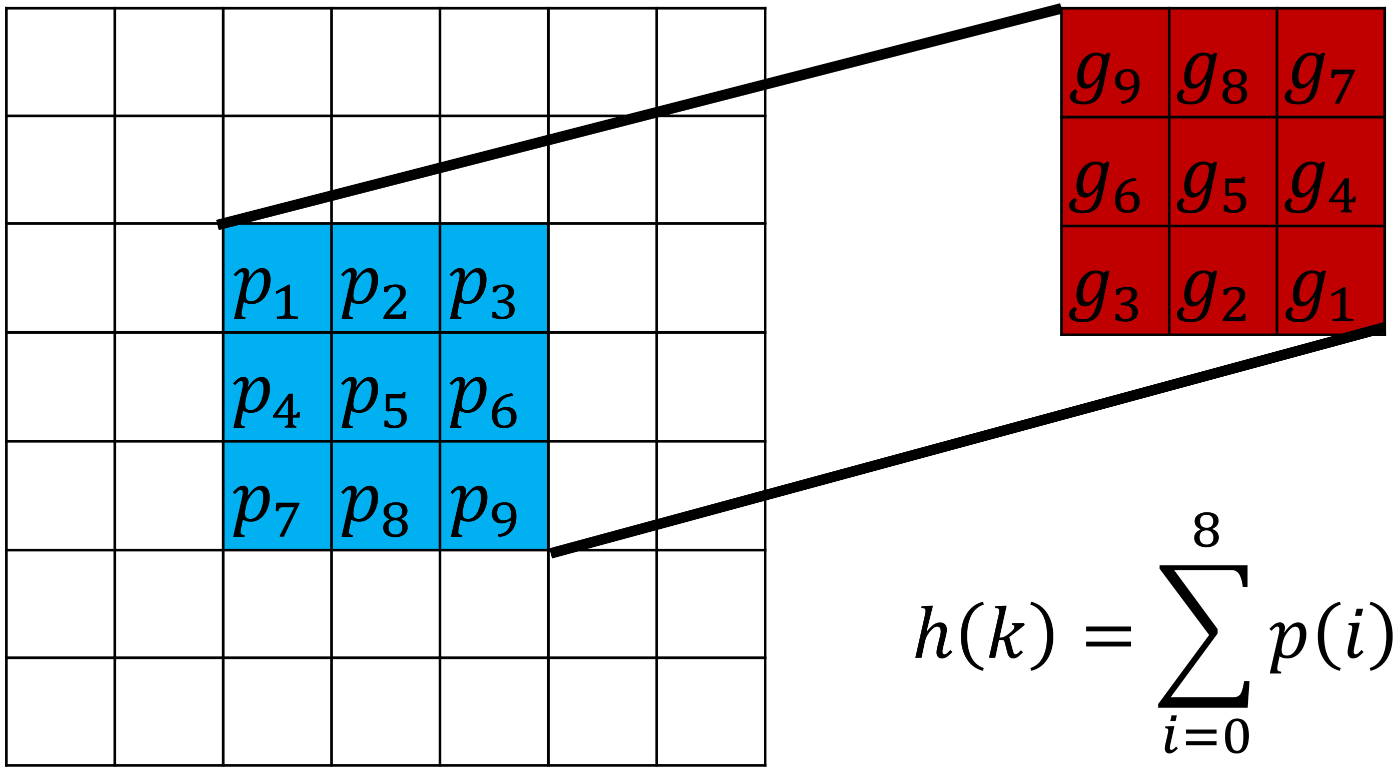
- **Why Reflect?**
  - Commutative: $f * g = g * f$
  - Associative: $f * (g * h) = (f * g) * h$
  - Distributive: $f * (g + h) = f * g + f * h$

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

2D Discrete Convolution

$$h = f * g$$

$$h(i,j) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} f(k,l)g(i-k, j-l)$$



$$h(k) = \sum_{i=0}^{8} p(i)g(i)$$

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Simple Convolutions

$$\begin{bmatrix}1 & 1\end{bmatrix}\begin{bmatrix}1\\1\end{bmatrix} = \begin{bmatrix}1 & 1\\1 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 1\end{bmatrix}\begin{bmatrix}1 & 1\end{bmatrix} = \begin{bmatrix}1 & 2 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 1\\1 & 1\end{bmatrix}\begin{bmatrix}1 & 1\\1 & 1\end{bmatrix} = \begin{bmatrix}1 & 2 & 1\\2 & 4 & 2\\1 & 2 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 1\end{bmatrix}^n$$

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

$$\begin{bmatrix}1 & 2 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 3 & 3 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 4 & 6 & 4 & 1\end{bmatrix}$$

$$\begin{bmatrix}1 & 2 & 1\end{bmatrix}\begin{bmatrix}1\\2\\1\end{bmatrix} = \begin{bmatrix}1 & 2 & 1\\2 & 4 & 2\\1 & 2 & 1\end{bmatrix}$$

Image Filtering

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Smoothing, Noise Cleaning

$$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Original Image

Noisy Image

Filtered Image

Noise Assumption: Additive, Independent

Smoothing, Noise Cleaning

Original

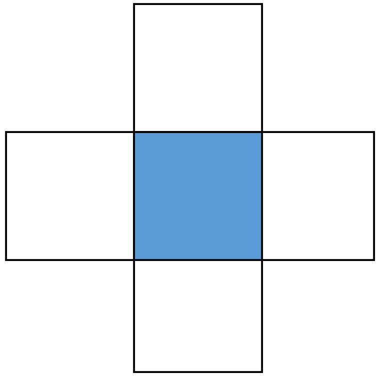
Smoothed

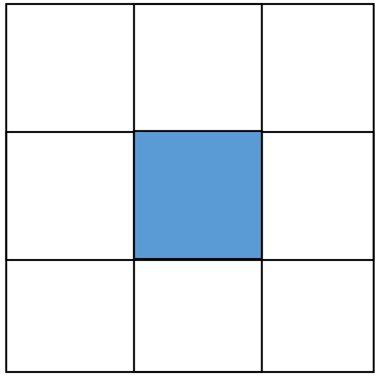
Smoothing by a larger kernel

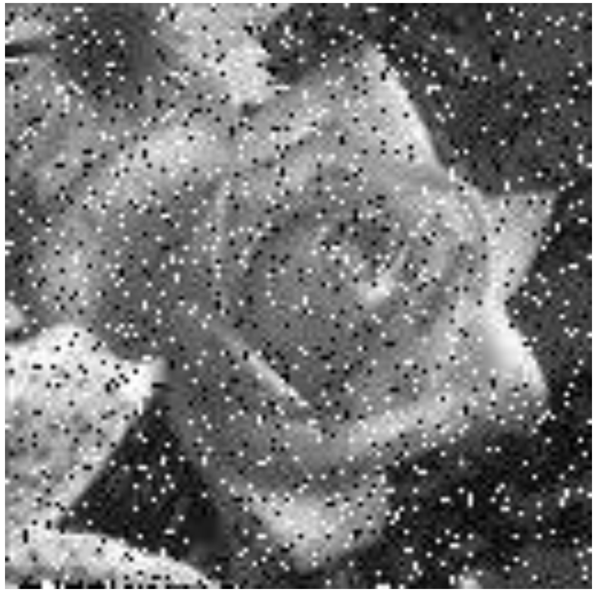
Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Noise Cleaning by Median Filtering

- Replace the value of a pixel with the MEDIAN of its neighborhood
- Depends on the definition of “neighborhood”



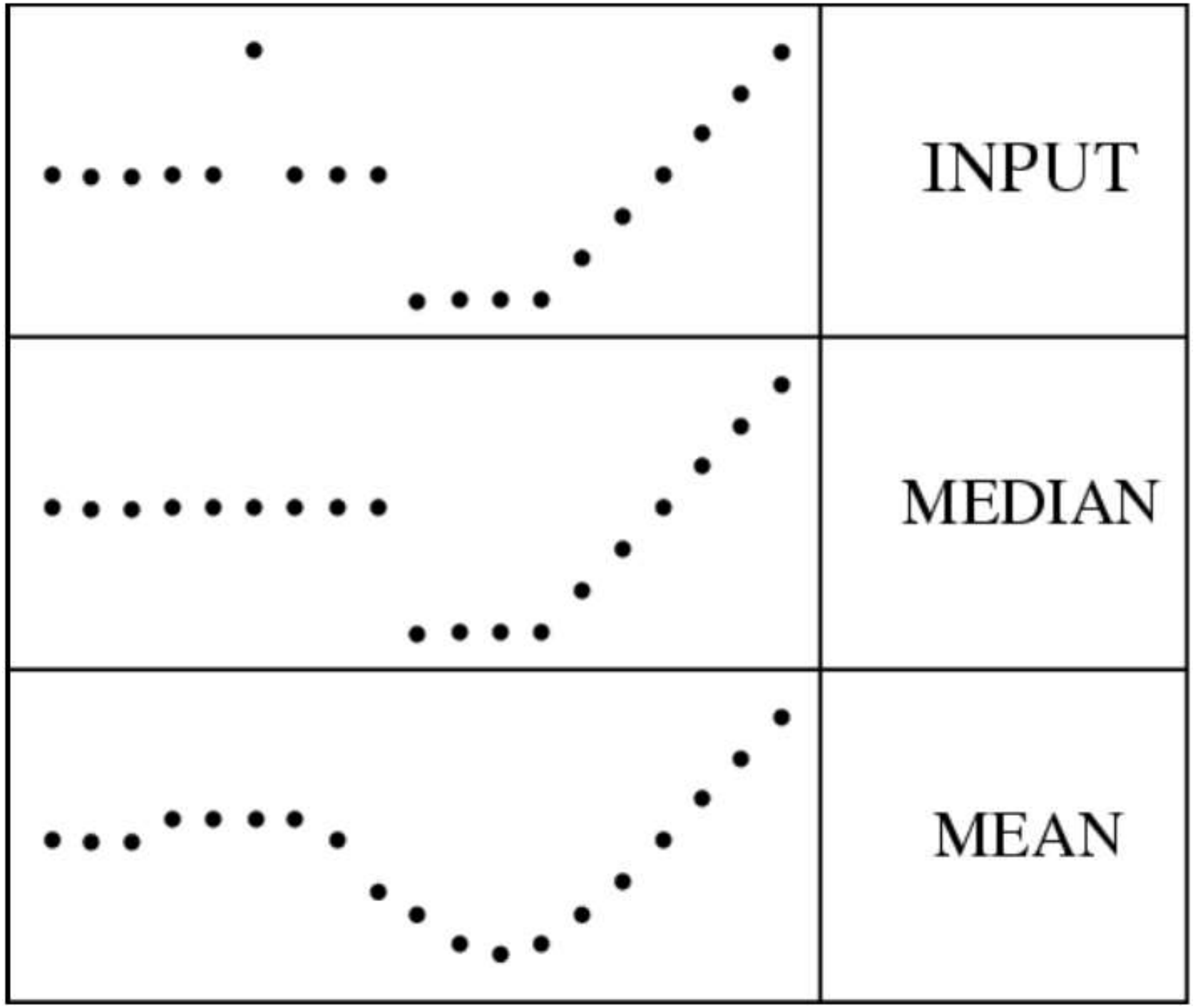




Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

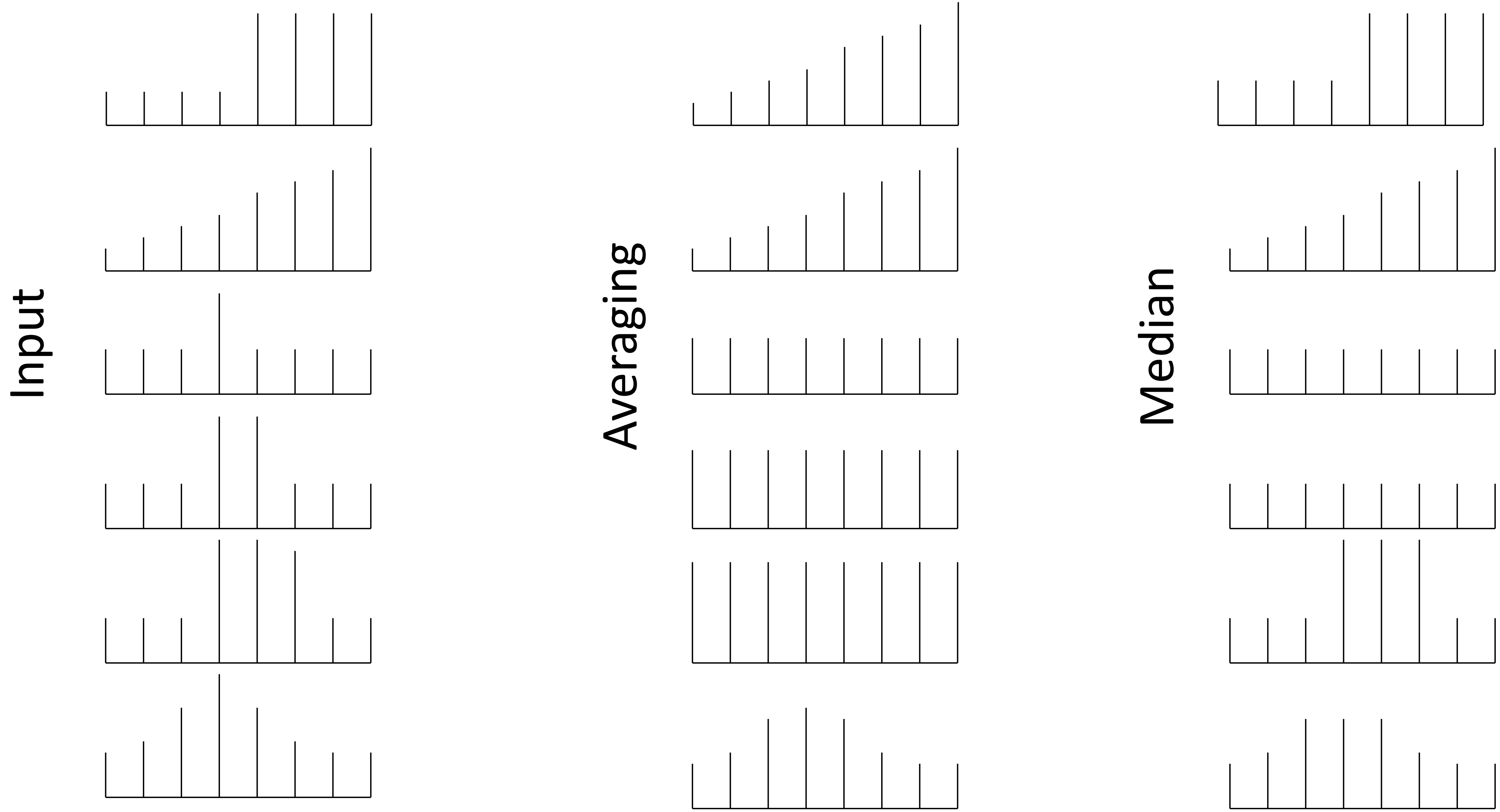
Average Vs Median Smoothing

Filter width = 5



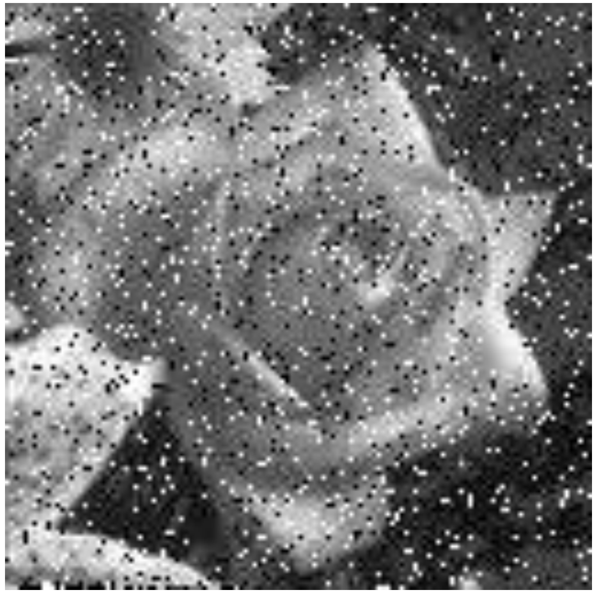
Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Average Vs Median Smoothing

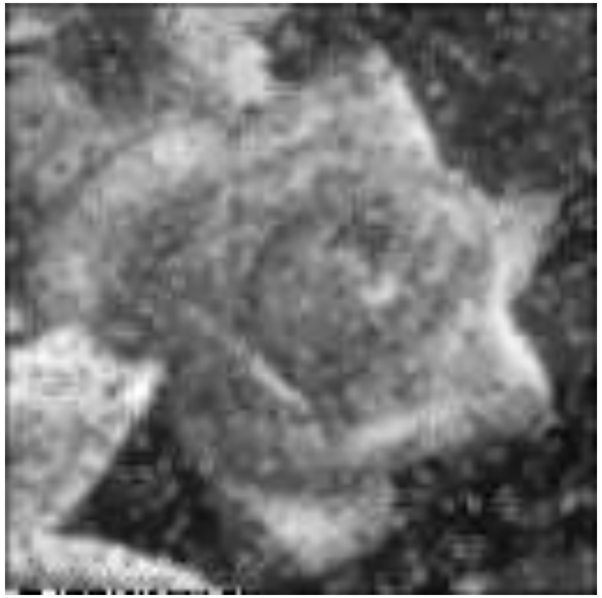


Average Vs Median Smoothing

- Averaging / Smoothing - Loss of Detail
- Median - Blockiness
- Min, Max etc.



Noisy Image
Salt & Pepper Noise



Averaging

Median

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Image Derivatives

$$\frac{\partial}{\partial x} f(i,j) = \lim_{\epsilon \to 0} \frac{f(i,j) - f(i-\epsilon, j)}{\epsilon}$$

$$\cong f(i,j) - f(i-1,j)$$

Implement

Convolution with $[1 \quad -1]$

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Image Derivatives

$$\frac{\partial}{\partial x} f(i,j) = \lim_{\epsilon \to 0} \frac{f(i,j) - f(i-\epsilon, j)}{\epsilon}$$

$$\cong f(i,j) - f(i-1,j)$$

Implement

Convolution with $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$

Types of Discrete derivative in 1D

- Backward: $\frac{\partial f}{\partial x} = f(x) - f(x-1) \Rightarrow [-1 \quad 1 \quad 0]$

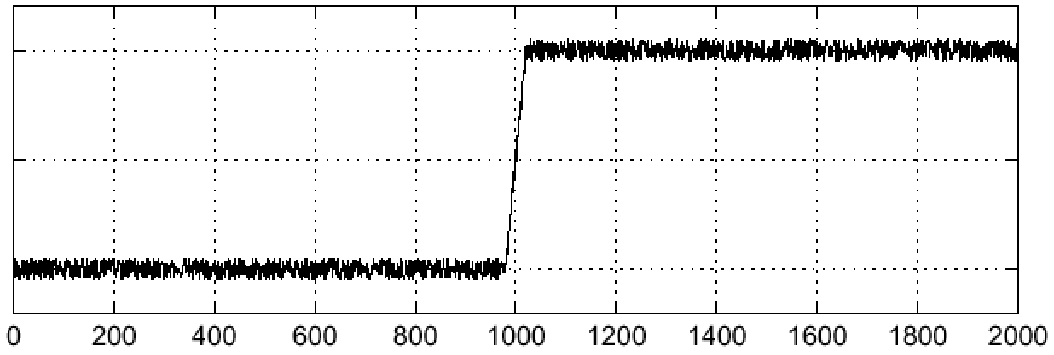
- Forward: $\frac{\partial f}{\partial x} = f(x) - f(x+1) \Rightarrow [0 \quad 1 \quad -1]$

- Central: $\frac{\partial f}{\partial x} = f(x+1) - f(x-1) \Rightarrow [-1 \quad 0 \quad +1]$

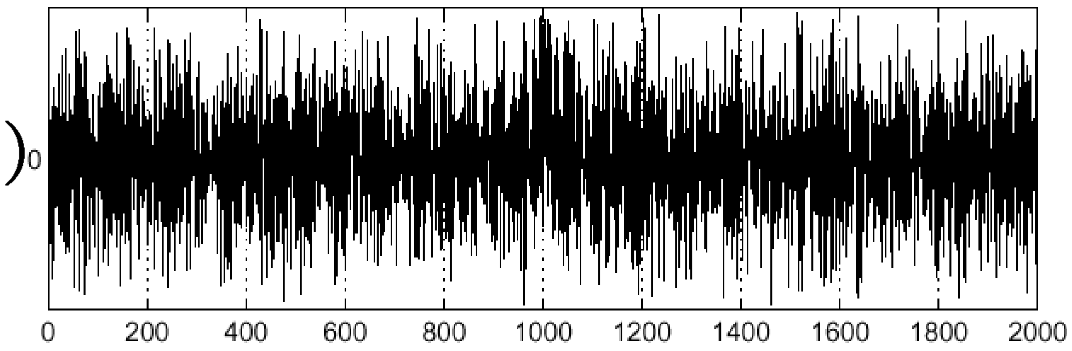
Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Effect of Noise on Image Gradients

$f(x)$



$\frac{d}{dx}f(x)$



Where is the edge?

Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Effect of Noise on Image Gradients

- Finite difference filters respond strongly to noise
  - Image noise results in pixels that look very different from their neighbors
  - Generally, the larger the noise the stronger the response

- What is to be done?
  - Smoothing the image should help, by forcing pixels different to their neighbors (=noise pixels?) to look more like neighbors

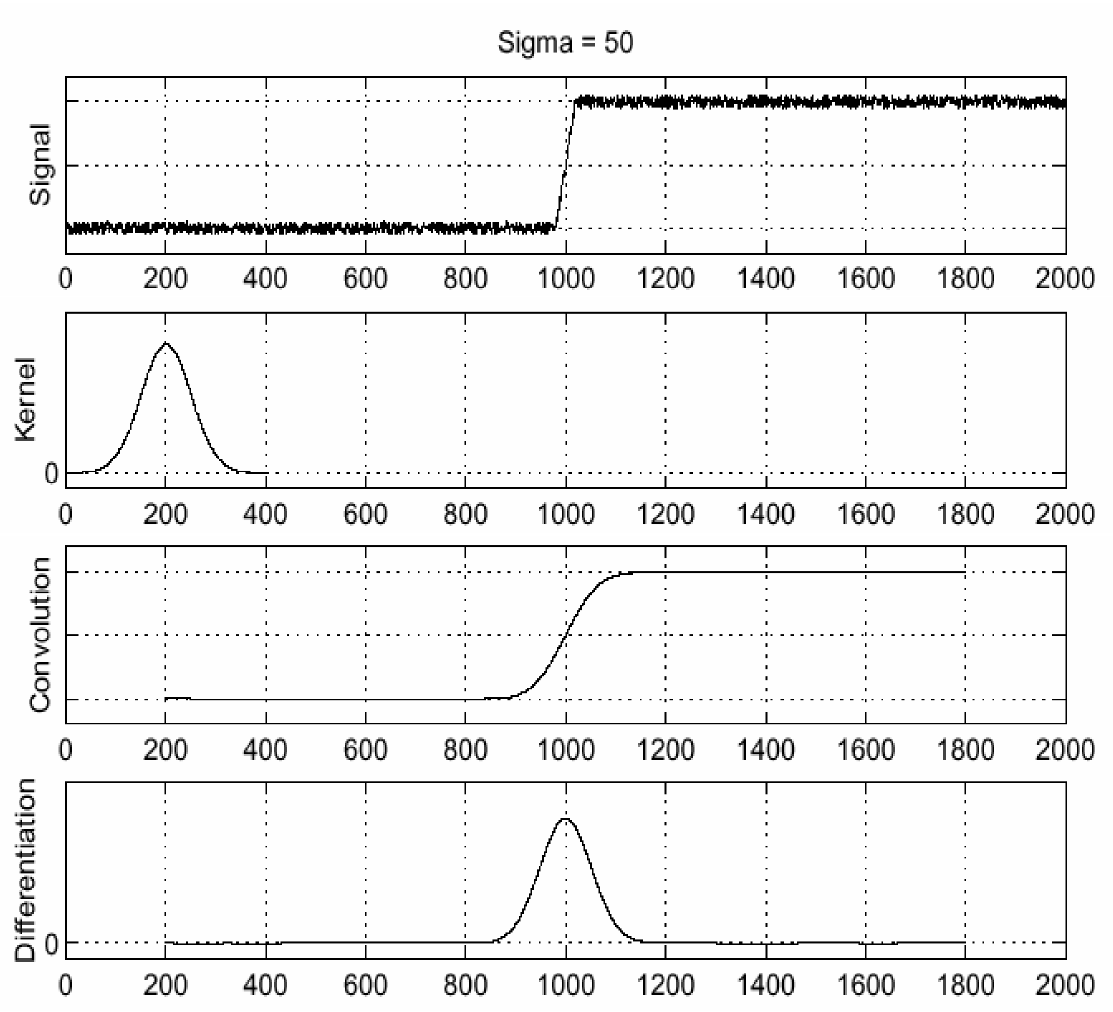
Effect of Noise on Image Gradients

f

g

$f * g$

$\frac{d}{dx}(f * g)$



Chetan Arora
Department of Computer Science and Engineering, IIT Delhi

Image Derivatives

- Good derivative filters are high-pass (edge) in one direction and low-pass (blur) in the orthogonal direction.

Sobel (X)

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

Sobel (Y)

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

- Derivative/Difference of Gaussians