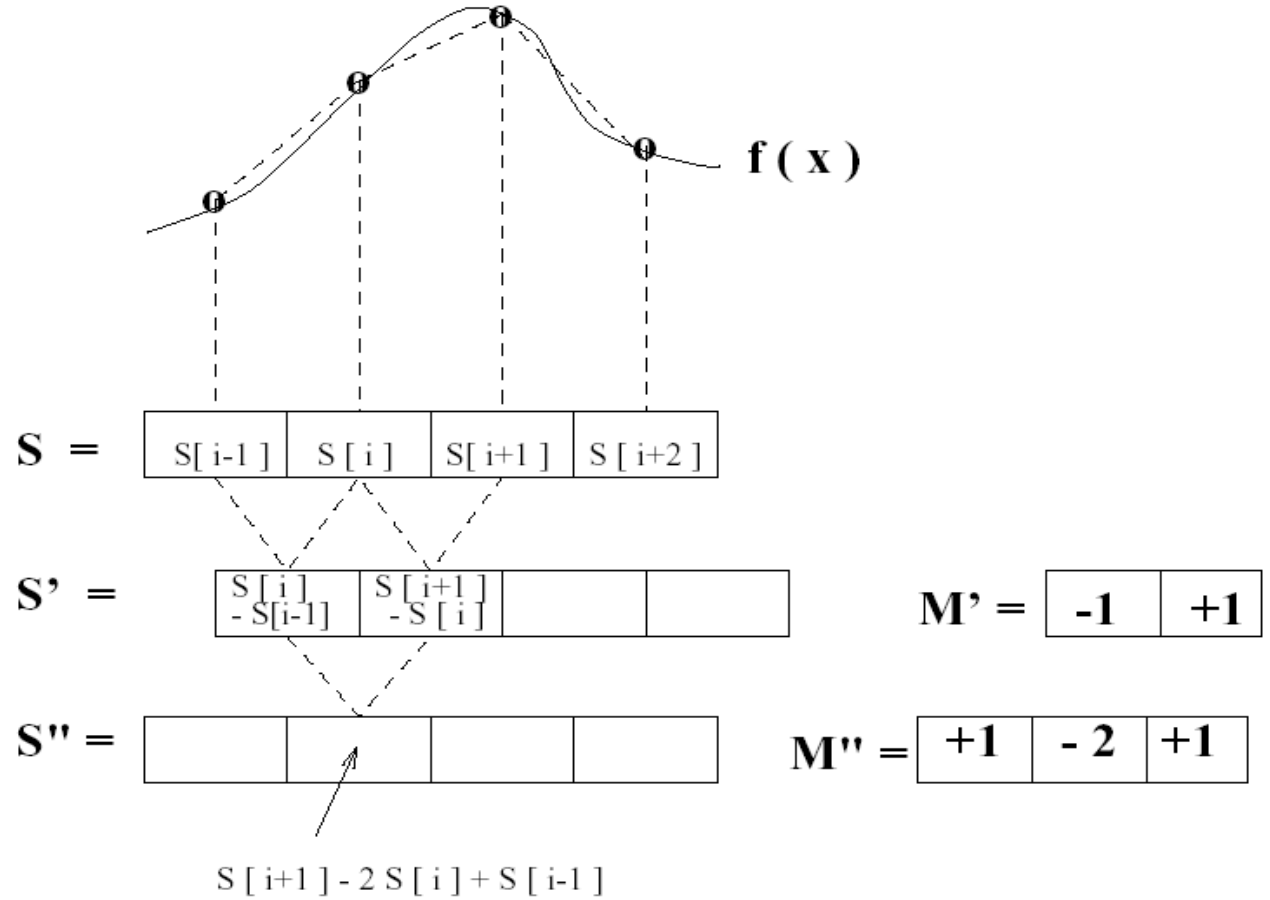
2. Basic Image Features

Eun Yi Kim

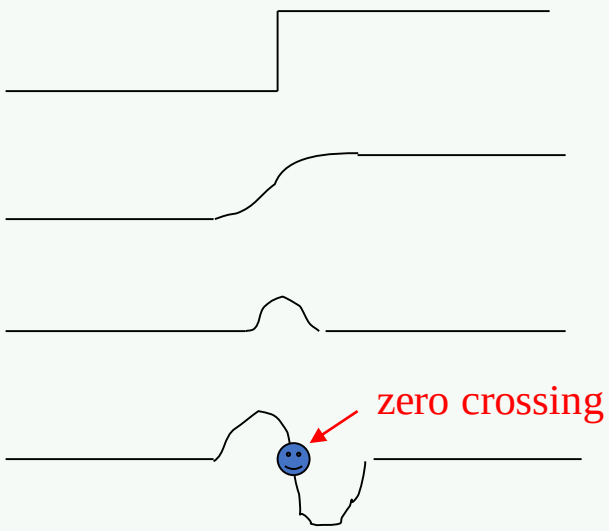
Artificial Inteligence
& Computer Vision
Laboratory

Differencing 1D Signals



Zero Crossing Operators

- Motivation: The zero crossings of the second derivative of the image function are more precise than the peaks of the first derivative.



step edge

smoothed

1st derivative

2nd derivative

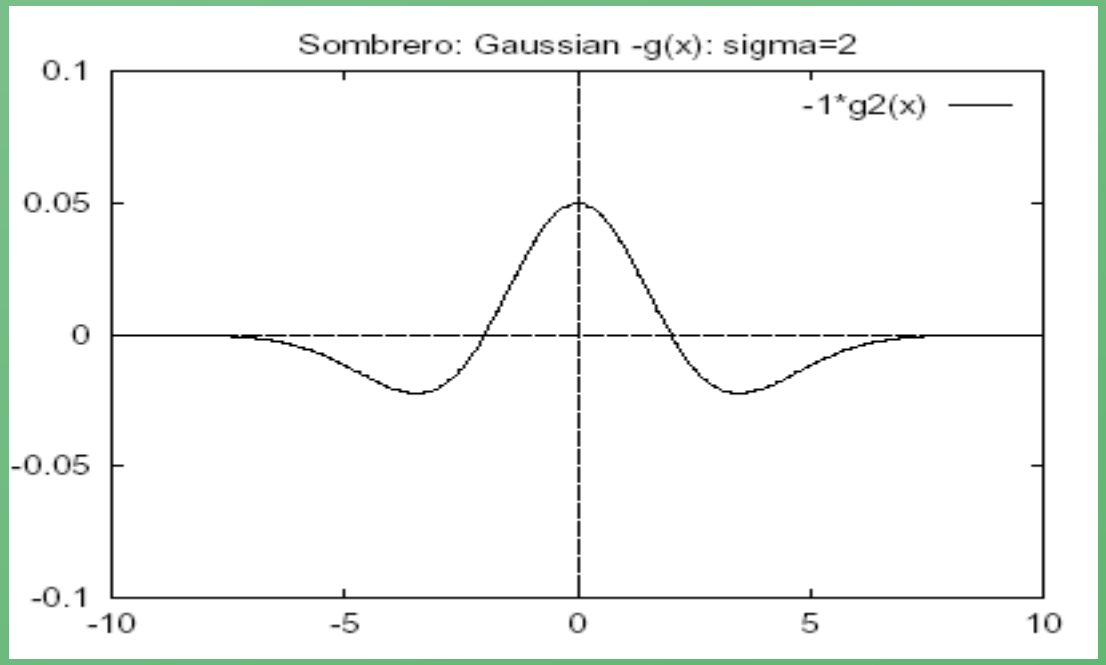
How do we estimate the Second Derivative?

- Laplacian Filter: $\nabla^2 f = \partial^2 f / \partial x^2 + \partial^2 f / \partial y^2$

0	1	0
1	-4	1
0	1	0

- Standard mask implementation
- Derivation: In 1D, the first derivative can be computed with mask [-1 0 1]
- The 1D second derivative is [1 -2 1]
- The Laplacian mask estimates the 2D second derivative.

Detecting Edges with Laplacian Operator



0	−1	0
−1	4	−1
0	−1	0

5	5	5	5	5	5
5	5	5	5	5	5
5	5	10	10	10	10
5	5	10	10	10	10
5	5	5	10	10	10
5	5	5	5	10	10

-	-	-	-	-	-
-	0	-5	-5	-5	-
-	-5	10	5	5	-
-	-5	10	0	0	-
-	0	-10	10	0	-
-	-	-	-	-	-

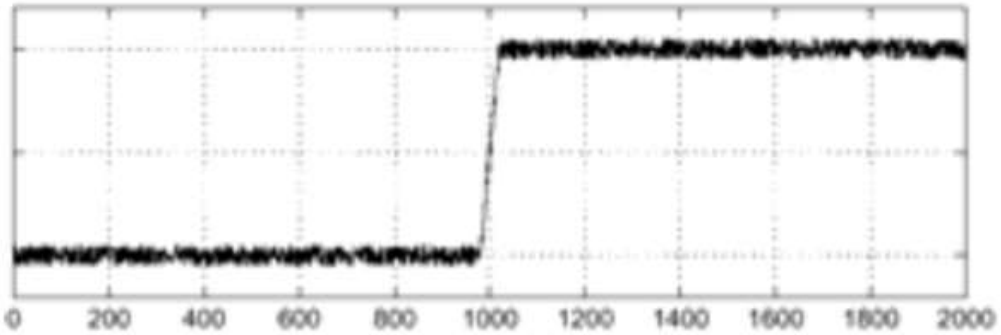
Edge Detection Background

- Classical gradient edge detection
  - Sobel, Prewitt, Kirsch and Robinson
- Zero-crossing based methods
  - Laplacian, LoG
- Gaussian based filters
  - Marr and Hildreth
  - Canny operator
- …

Effect of Noise

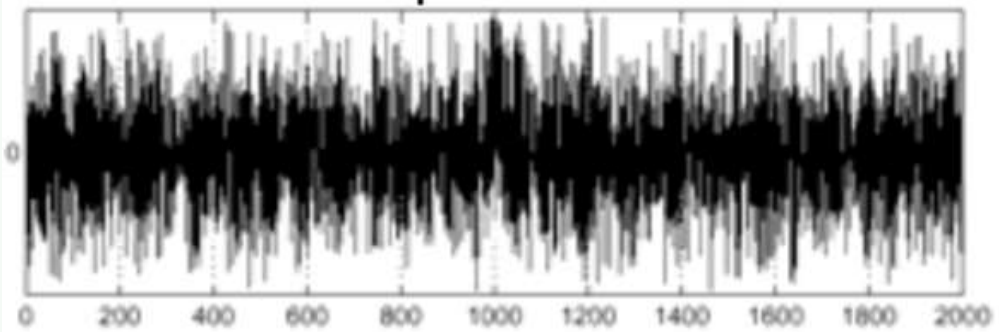
- Consider a single row or column of the image
  - Plotting intensity as a function of position gives a signal

$f(x)$



How to compute a derivative?

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



- Where is the edge?

Effect of Noise

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

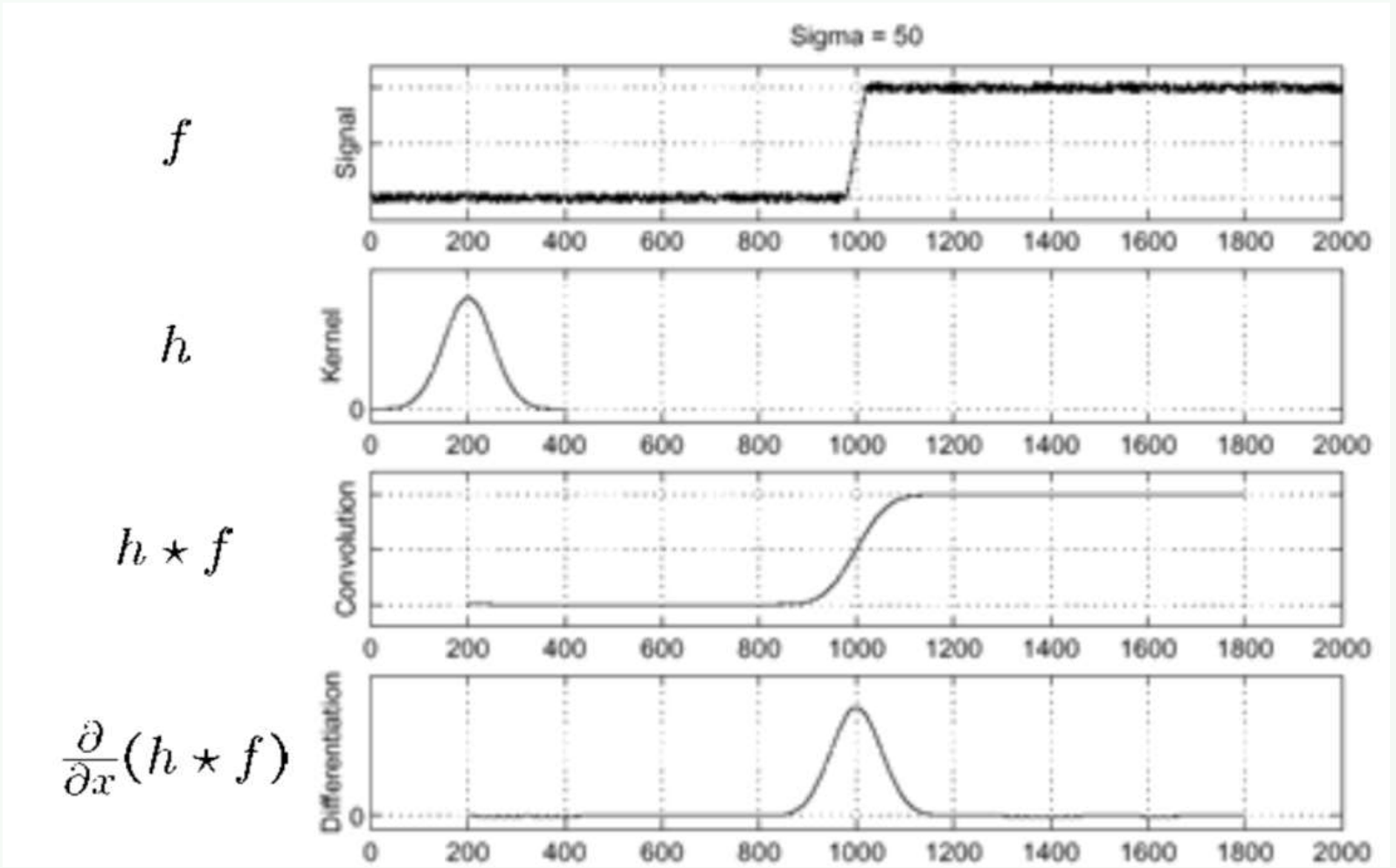
- What is to be done?

Effect of Noise

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

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

Solution: smooth first

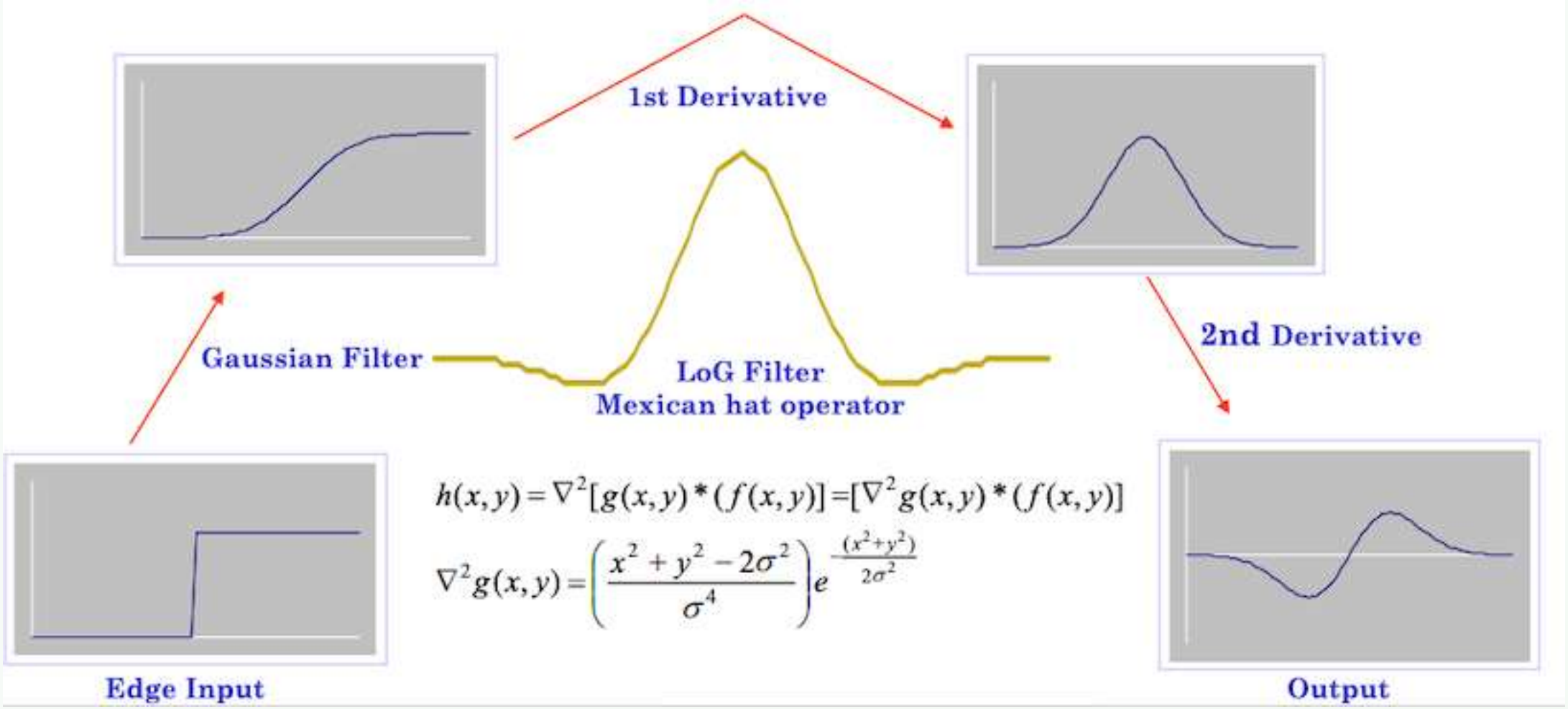


- Where is the edge?
  - Look for peaks

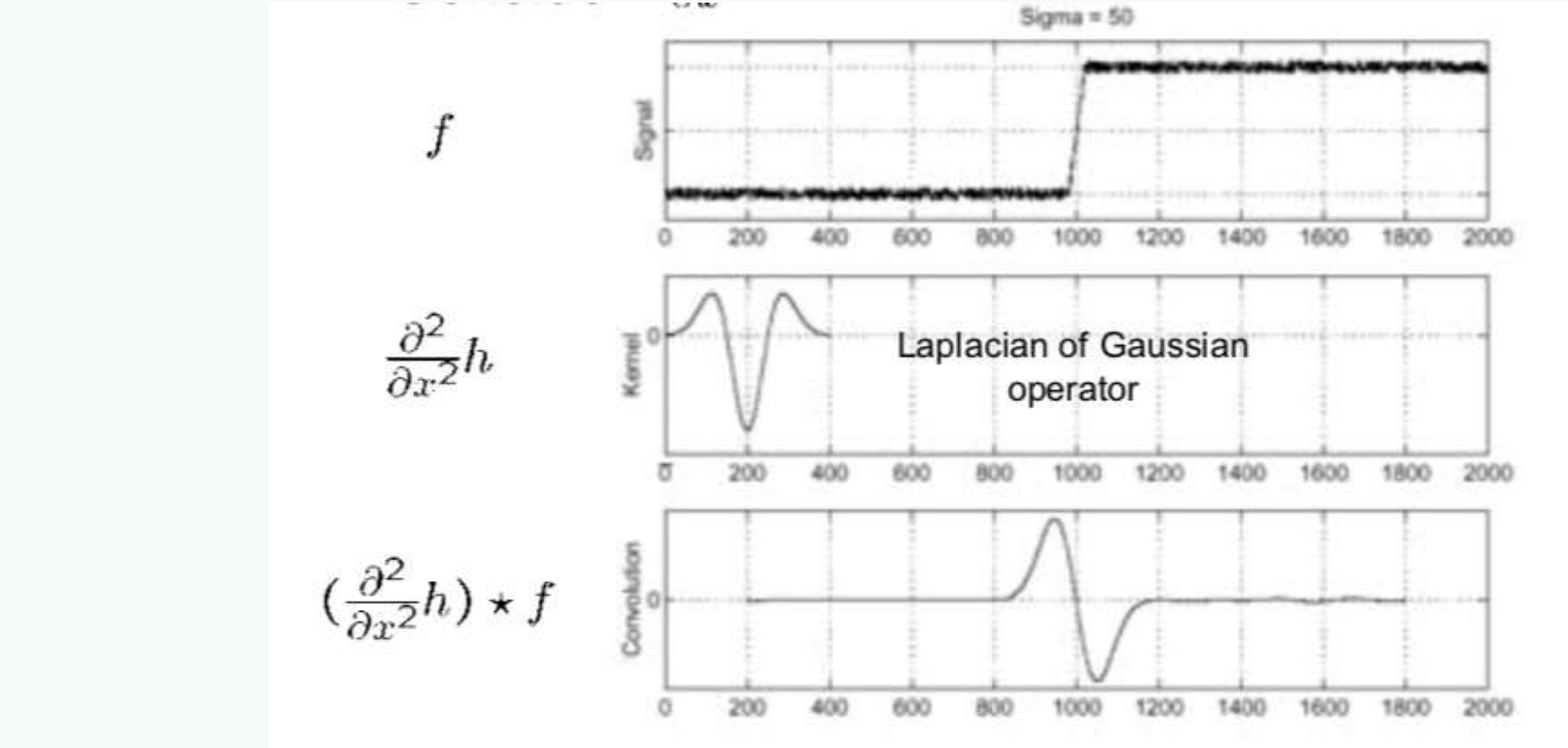
Laplacian of Gaussian (LoG) : Marr and Hildreth Operator

- First smooth the image via a Gaussian convolution.
- Apply a Laplacian filter (estimate 2nd derivative).
- Find zero crossings of the Laplacian of the Gaussian.
  - Only the zero crossings whose corresponding 1st derivative is above a specified threshold are considered
- Edge location can be estimated with subpixel resolution using linear interpolation

Laplacian of Gaussian (LoG)



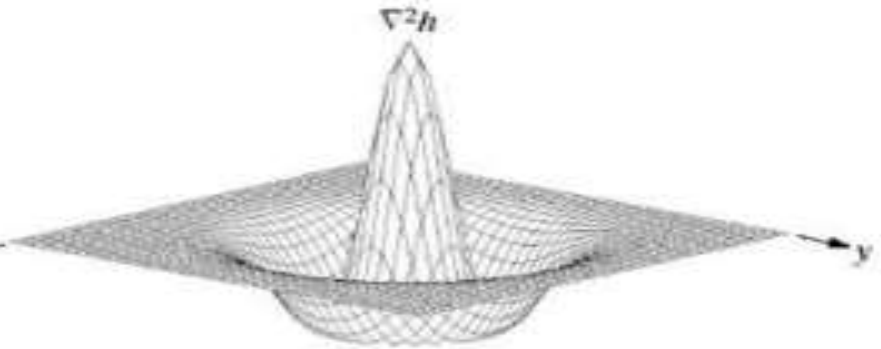
Laplacian of Gaussian (LoG)

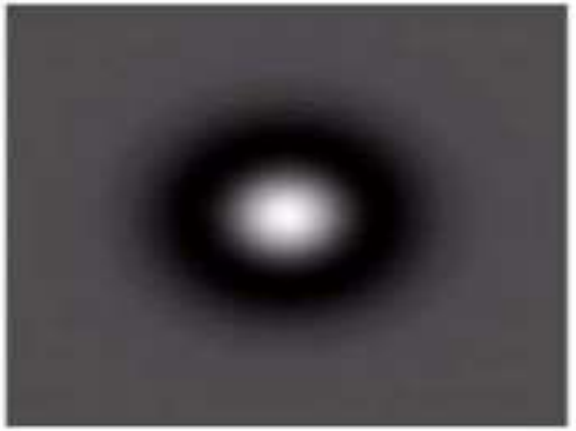


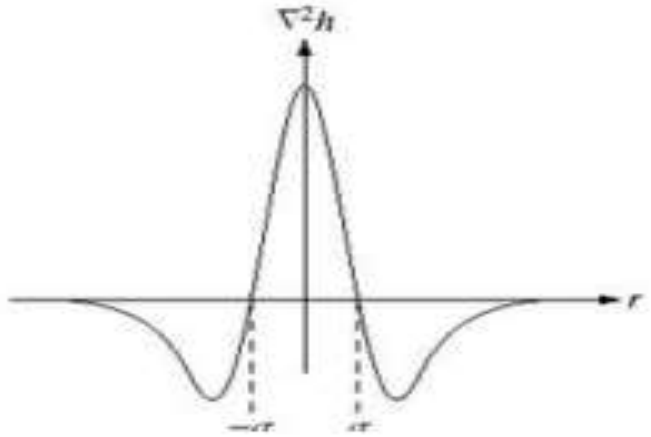
- Where is the edge?
  - Zero-crossing of bottom graph

Laplacian of Gaussian (LoG)

Inverted LoG function (Mexican hat)







0	0	−1	0	0
0	−1	−2	−1	0
−1	−2	16	−2	−1
0	−1	−2	−1	0
0	0	−1	0	0

Laplacian of Gaussian (LoG)

Scale space

5 x 5 LoG filter

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

17 x 17 LoG filter

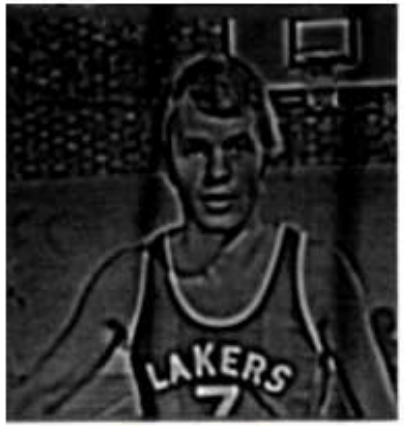
0	0	0	0	0	0	-1	-1	**-1**	-1	-1	0	0	0	0	0
0	0	0	0	-1	-1	-1	-1	**-1**	-1	-1	-1	-1	0	0	0
0	0	-1	-1	-1	-2	-3	-3	**-3**	-3	-3	-2	-1	-1	-1	0
0	0	-1	-1	-2	-3	-3	-3	**-3**	-3	-3	-3	-2	-1	-1	0
0	-1	-1	-2	-3	-3	-3	-2	**-3**	-2	-3	-3	-3	-2	-1	-1
0	-1	-2	-3	-3	-3	0	2	**4**	2	0	-3	-3	-3	-2	-1
-1	-1	-3	-3	-3	0	4	10	**12**	10	4	0	-3	-3	-3	-1
-1	-1	-3	-3	-2	2	10	18	**21**	18	10	2	-2	-3	-3	-1
-1	**-1**	**-3**	**-3**	**-3**	**4**	**12**	**21**	**24**	**21**	**12**	**4**	**-3**	**-3**	**-3**	**-1**
-1	-1	-3	-3	-2	2	10	18	**21**	18	10	2	-2	-3	-3	-1
-1	-1	-3	-3	-3	0	4	10	**12**	10	4	0	-3	-3	-3	-1
0	-1	-2	-3	-3	-3	0	2	**4**	2	0	-3	-3	-3	-2	-1
0	-1	-1	-2	-3	-3	-3	-2	**-3**	-2	-3	-3	-3	-2	-1	-1
0	-1	-1	-2	-3	-3	-3	-2	**-3**	-2	-3	-3	-3	-2	-1	-1
0	0	-1	-1	-1	-2	-3	-3	**-3**	-3	-3	-2	-1	-1	-1	0
0	0	0	0	-1	-1	-1	-1	**-1**	-1	-1	-1	-1	0	0	0

Scale (σ)

Laplacian of Gaussian (LoG)

Scale space

Original Image



LoG Filter

Zero Crossings

Scale (σ)

Edge Detection Results

Original gray scale

Additive Gaussian Noise

Edge Detection Results

- Roberts operator
  - Poor robustness to noise, low detection

Edge Detection Results

- Sobel operators
  - Better robustness to noise, better detection

Edge Detection Results

- LoG operator
  - Better robustness to noise, better detection

Implementation issues

- The gradient magnitude is large along a thick “trail” or “ridge”, so how do we identify the actual edge points?
- How do we link the edge points to form curves?

Canny Edge Operators on Kidney