COMP 558

Lecture 4

Edge Detection

Linear Systems

- A **linear** system satisfies the properties of additivity and homogeneity.
- **Additivity**: If $I_1(x)$ as input gives $O_1(x)$ as output and $I_2(x)$ as input gives $O_2(x)$ as output then $I_1(x) + I_2(x)$ as input gives $O_1(x) + O_2(x)$ as output.
- **Homogeneity**: If $I_1(x)$ as input gives $O_1(x)$ as output then the input $a.I_1(x)$ gives $a.O_1(x)$ as output. Here "a" is a complex constant and "." denotes scalar multiplication.
- These two properties together are referred to as **superposition**.

Shift (or Time) Invariance

- A linear system is said to be shift (or time) invariant if the shape of its output does not change when it is given the same input at a different time.
- Mathematically what this means is that if $O_1(x)$ is the output when $I_1(x)$ is the input, then, given as input $I_1(x\text{-}u)$, the output is $O_2(x)=O_1(x\text{-}u)$.

The Convolution Theorem

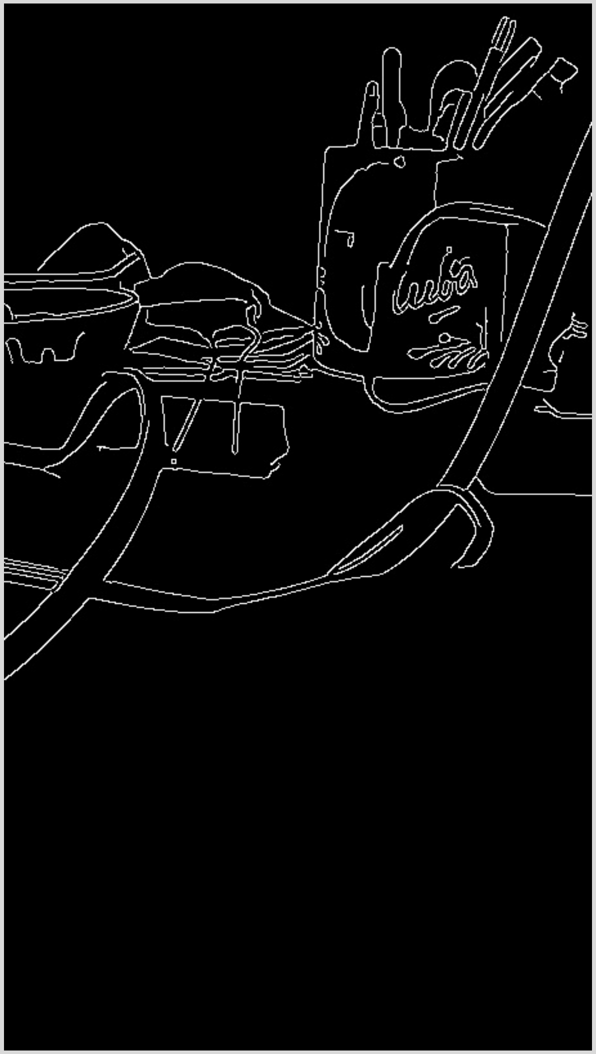
- A beautiful property of a linear shift invariant system is that it is completely characterized by its impulse response!
- Let h(x) be the recorded output of the system given an input δ(x) as input (the impulse response).
- Then, its output O(x) for any abitrary input I(x) is given by

$$h(x)*I(x) \equiv \sum_{u} h(x-u)\, I(u)$$

Example

Input image

Output of “Canny” edge detector



Motivation for edge detection

- Find boundaries of objects
  - recognize shapes based on their 2D silhouettes
  - estimate scene layout
- Match image "features" from one image to the next
  - binocular vision (stereo)
  - image motion (video)

Today: classical edge detection

- Prewitt and Sobel edge detectors (1960’s)
- Marr & Hildreth edge detector (1979)
- Canny edge detector (1986)

Edge detection based on a threshold

Consider a filter [1, 0, -1] defined on x = {-1, 0, 1} .

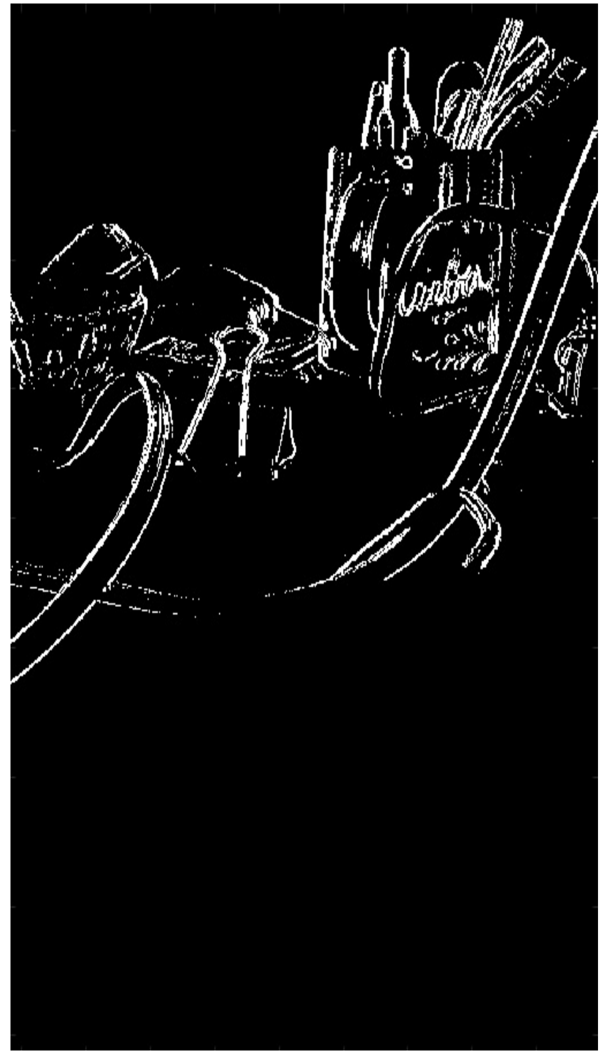
Given an image $I(x)$, find x such that

$$| \ I(x+1) - \ I(x-1) \ | \ > \quad \tau$$

where τ is some arbitrary *threshold*.

What if we apply edge detection method (previous slide) *within each row* ?

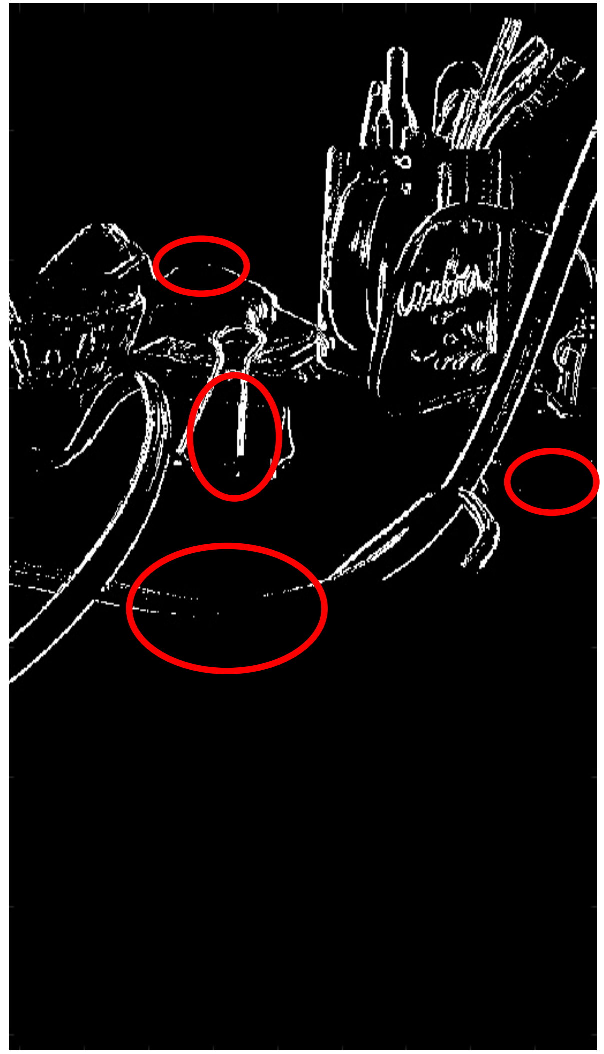
Input image



Output of naïve 1D edge detector (threshold = 100).

What if we apply edge detection method (previous slide) *within each row* ?

Input image



Output of naïve 1D edge detector (threshold τ = 100).

Missing horizontal edges !

Also, the edges are thick (many pixels wide). Thickness depends on the threshold τ.

Recall from last lecture: Image gradient

$$\nabla I(x, y) \equiv \left(\frac{\partial}{\partial x} I(x, y), \frac{\partial}{\partial y} I(x, y) \right)$$

$$\approx \left(\frac{1}{2} I(x+1, y) - \frac{1}{2} I(x-1, y), \qquad \frac{1}{2} I(x, y+1) - \frac{1}{2} I(x, y-1) \right)$$

There are many ways to define gradient-like kernels (filters). e.g. Rather than using 1x 3 and 3x1 kernels for the partial derivative, we could use 3x3 kernels.

$$\frac{\partial\, I(x,y)}{\partial x} \approx I(x,y) * \begin{array}{|c|c|c|} \hline 1 & 0 & -1 \\ \hline 1 & 0 & -1 \\ \hline 1 & 0 & -1 \\ \hline \end{array}$$

$$\frac{\partial\, I(x,y)}{\partial y} \approx I(x,y) * \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline 0 & 0 & 0 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

$$\| \nabla I(x,y) \| \equiv \sqrt{\left(\frac{\partial}{\partial x} I(x,y)\right)^2 + \left(\frac{\partial}{\partial y} I(x,y)\right)^2}$$

Substitute values computed on previous slide

As before, given an image $I(x,y)$, find locations (x,y) such that

$$\| \nabla I(x,y) \| \quad > \quad \tau$$

where τ is an arbitrary threshold.

“Prewitt” edge detection (1970) estimates gradient using kernels:

(as on previous slides)

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

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

“Sobel” edge detection (1968) estimates gradient using kernels:

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

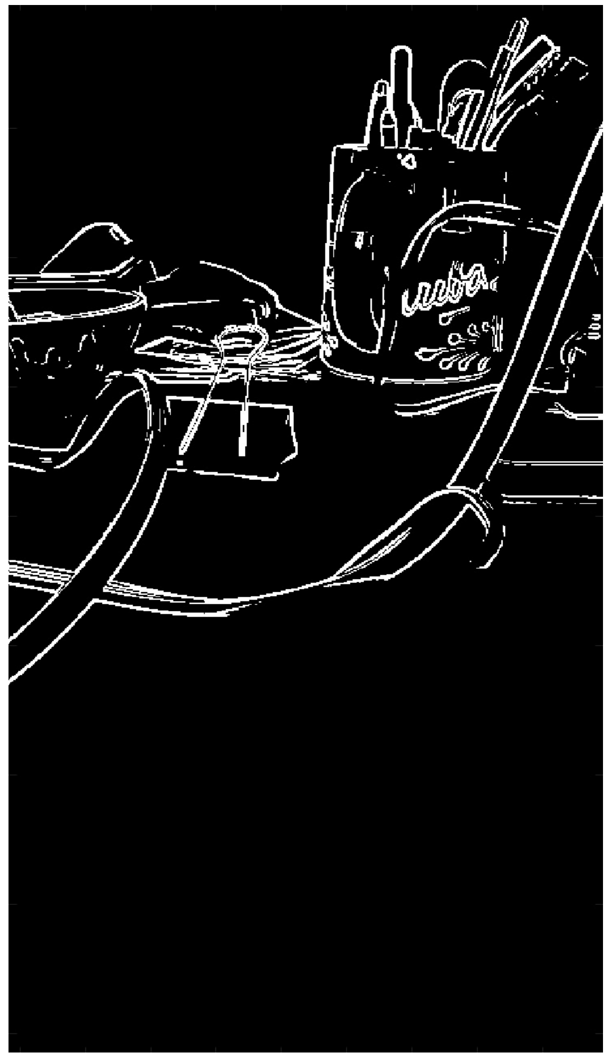
-1	-2	-1
0	0	0
1	2	1

ASIDE: To estimate gradient as “change in intensity per pixel distance”, one really should multiply by some constant i.e. normalize properly.

Example

Input image

Output of Prewitt edge detector



As before, we have marked points whose gradient magnitude exceeds some threshold.

Note that:

- edges of all orientations are found.

- edges are noisy relative to the Canny example earlier

Two issues to deal with:

1. Edges should be allowed to have any orientation.

2. Edges should be thin (preferably one pixel wide).

We will examine two methods that both deal with these issues:

- Marr and Hildreth (1979)

- Canny (1986)

Second Derivative

$$\frac{d^2\, I(x)}{dx^2} \approx I(x+1) - I(x) - \big(I(x) - I(x-1)\big)$$

Forward difference:
Estimate of derivative
at $x + \frac{1}{2}$

Backwards difference:
Estimate of derivative
at $x - \frac{1}{2}$

$$= I(x+1) - 2\, I(x) + I(x-1)$$

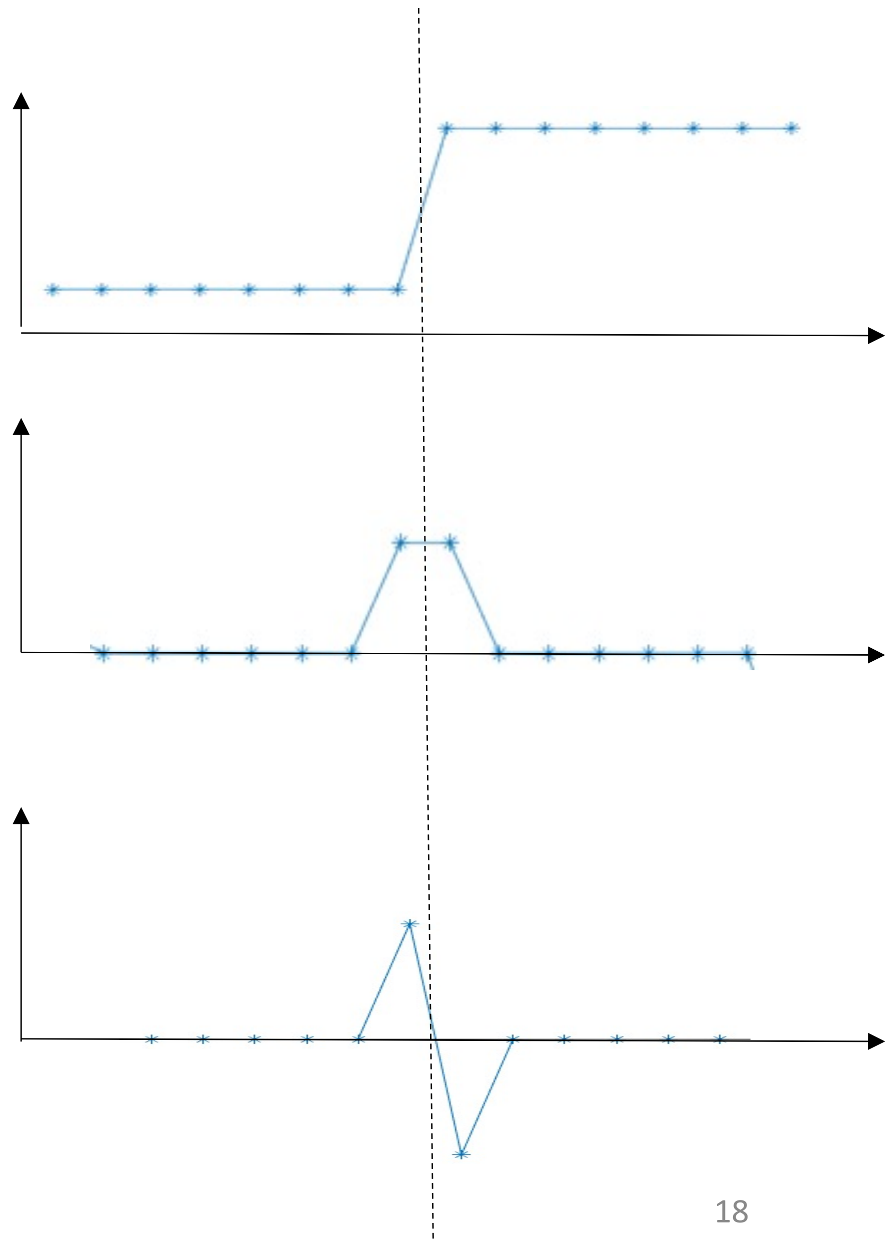
1	-2	1

Example: step edge

$I(x)$

$$\frac{d\ I(x)}{dx} \approx \frac{1}{2} I(x+1) - \frac{1}{2} I(x-1)$$

$$\frac{d^2\ I(x)}{dx^2} \approx I(x+1) - 2\,I(x) + I(x-1)$$

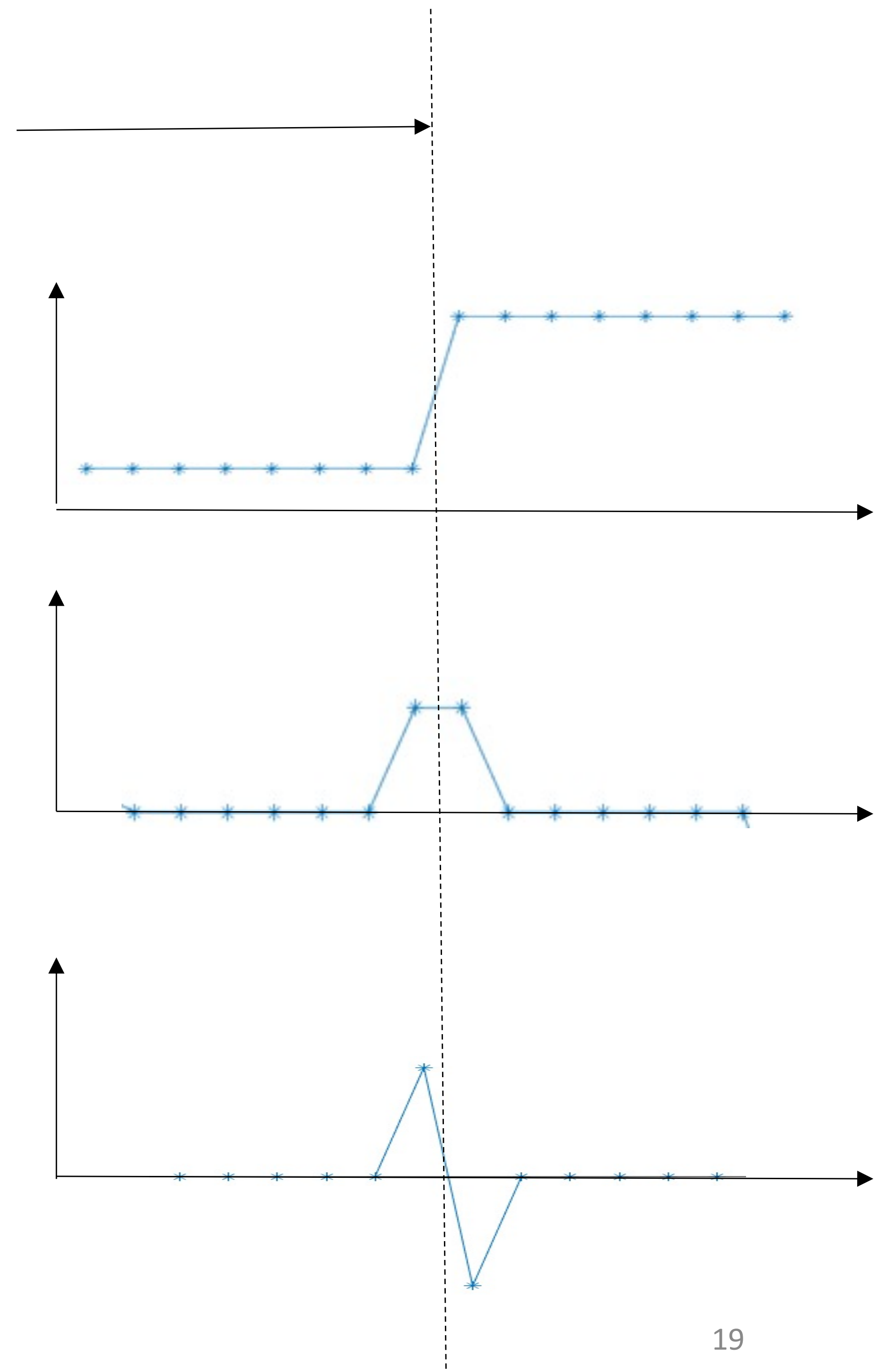


Dashed line marks the alignment of:

$I(x)$ Edge

$\dfrac{d\, I(x)}{dx}$ Maximum of first derivative

$\dfrac{d^2\, I(x)}{dx^2}$ Zero-crossings of second derivative



Second derivatives will be sensitive to image noise, so we will need to smooth the image to reduce noise.

If we smooth the image, we also smooth the edges.

We will use Gaussian smoothing, with standard deviation σ, since this provides a convenient way of adjusting the amount of smoothing.

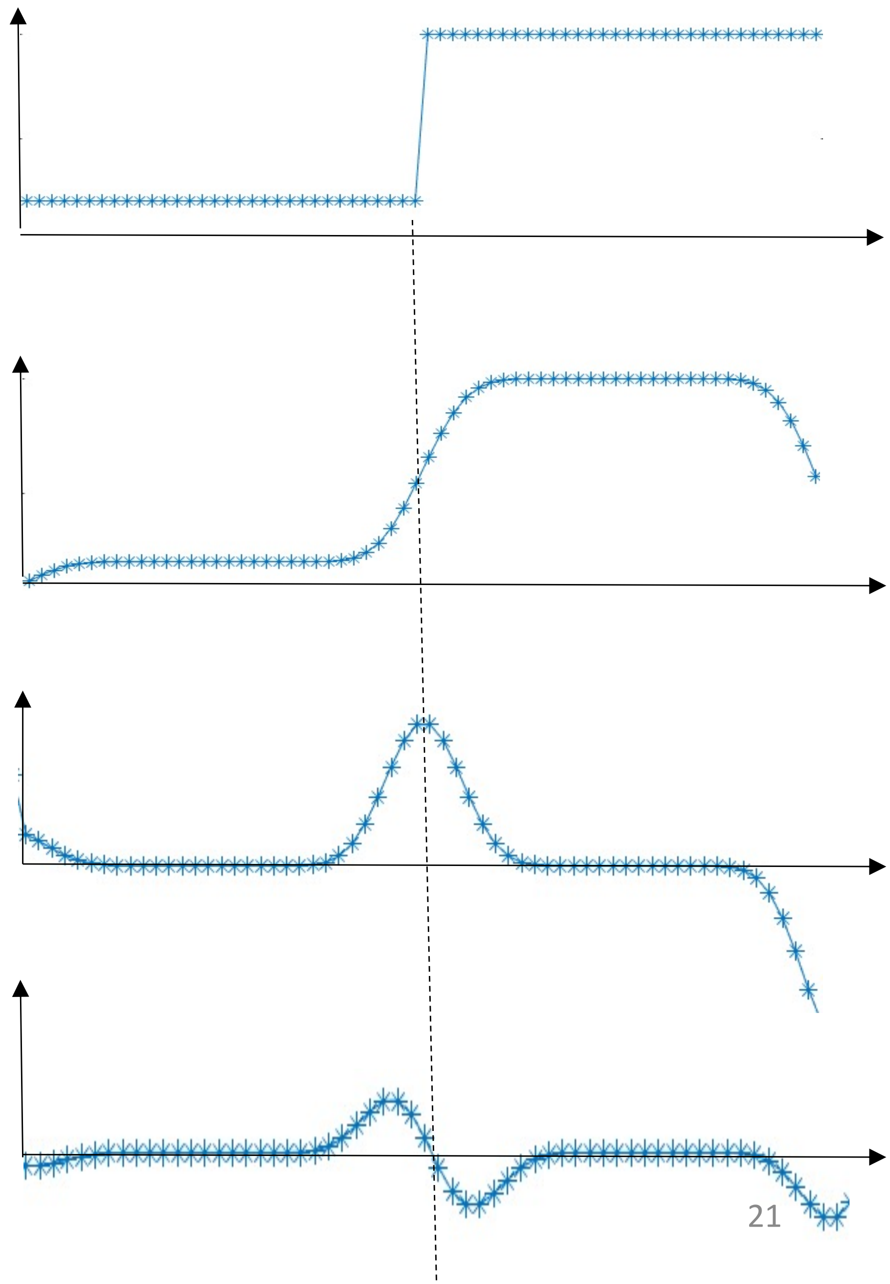
Example: blurred step edge (no noise)

$$I(x)$$

$$G(x,\sigma) * I(x)$$

$$\frac{d\, G(x,\sigma) * \; I(x)}{dx} \approx$$

$$\frac{d^2 G(x,\sigma) * I(x)}{dx^2} \approx$$



Recall the following property of convolution (associativity) which holds for both continuous and discrete functions:

$$f_1 * (f_2 * I) = (f_1 * f_2) * I$$

In particular, one can show:

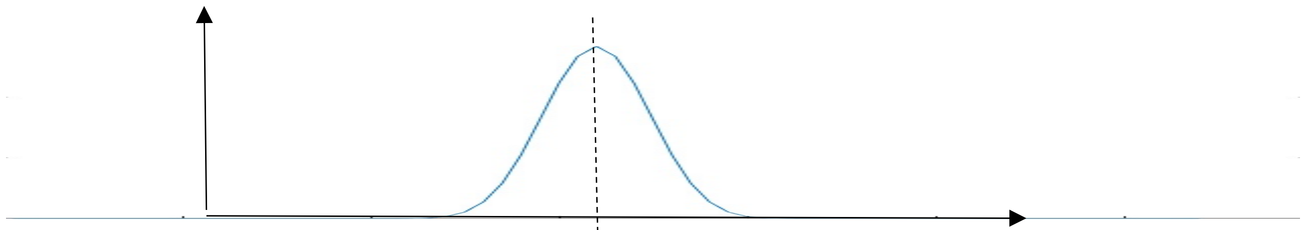
$$\frac{d}{dx}\big(G(x,\sigma) * I(x)\big) = \frac{d\, G(x,\sigma)}{dx} * I(x)$$

That is, a derivative can be written as a convolution.

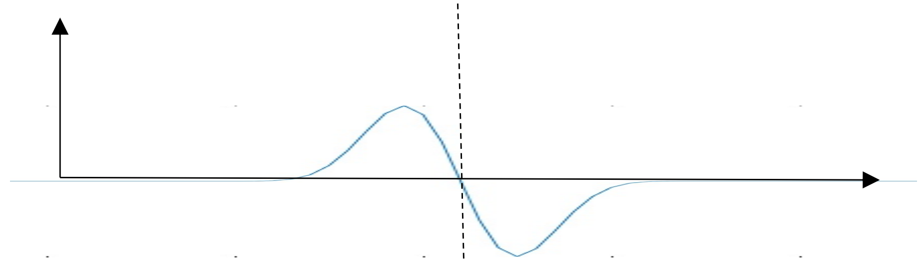
For discrete (sampled) functions, the derivative is approximated by a local difference.

Derivative of Gaussian

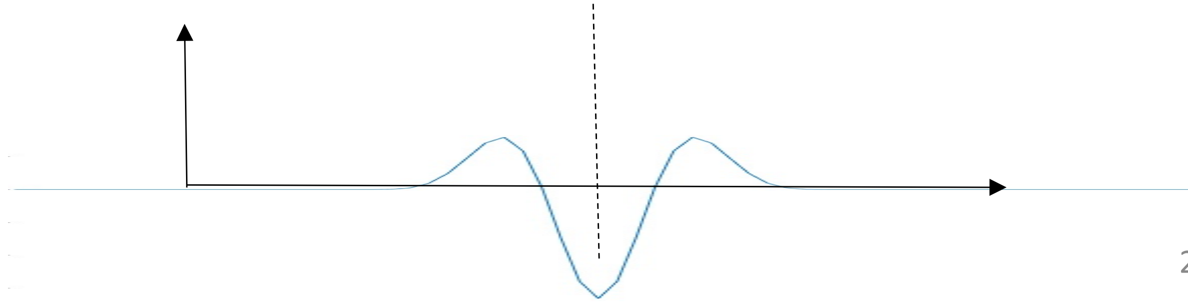
$$G(x, \sigma)$$



$$\frac{d\, G(x, \sigma)}{dx}$$

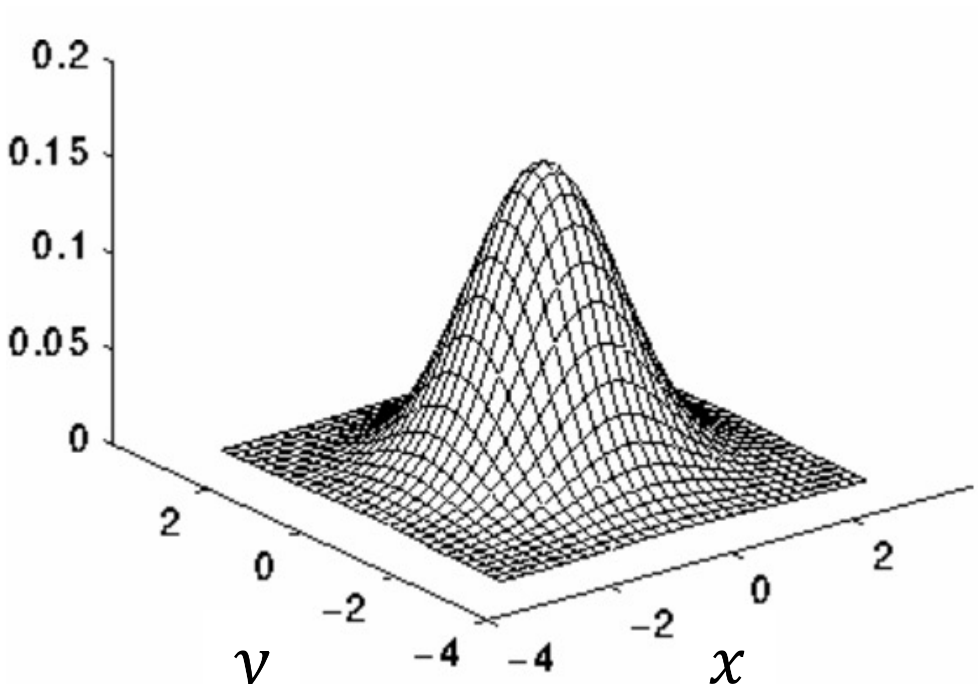


$$\frac{d^2\, G(x, \sigma)}{d\, x^2}$$



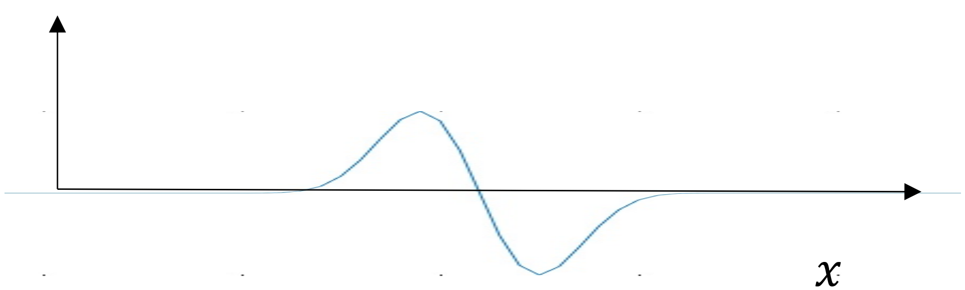
Recall 2D Gaussian

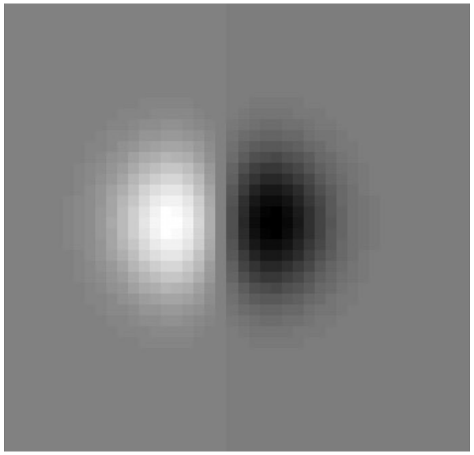
$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}}$$



2D Derivative of Gaussian

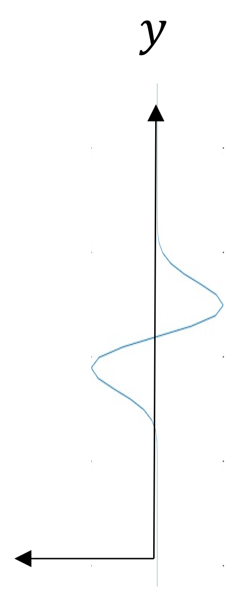
$$\frac{\partial G(x, y, \sigma)}{\partial x} \approx \begin{array}{|c|c|c|}\hline 1 & 0 & -1 \\ \hline\end{array} * G(x, y, \sigma)$$

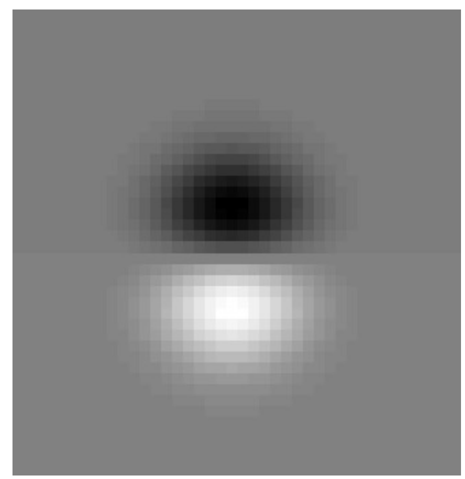




2D Derivative of Gaussian

$$\frac{\partial G(x,y,\sigma)}{\partial y} \approx \begin{array}{|c|}\hline -1 \\ \hline 0 \\ \hline 1 \\ \hline \end{array} * G(x,y,\sigma)$$





Gradient of smoothed image

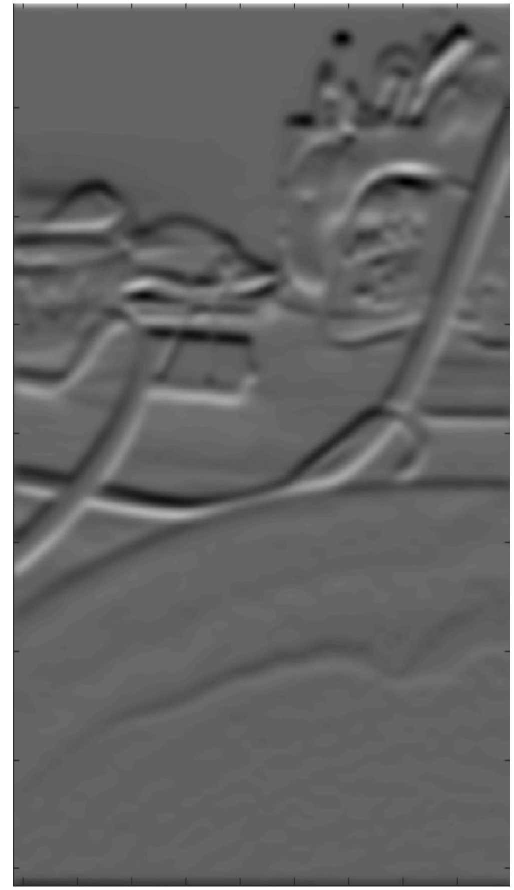
$$\nabla\, G(x, y, \sigma) * \; I(x, y)$$

$$\equiv \left(\frac{\partial}{\partial x} G(x, y, \sigma) * \; I(x, y), \qquad \frac{\partial}{\partial y} G(x, y, \sigma) * I(x, y) \right)$$

Example

$I(x, y)$

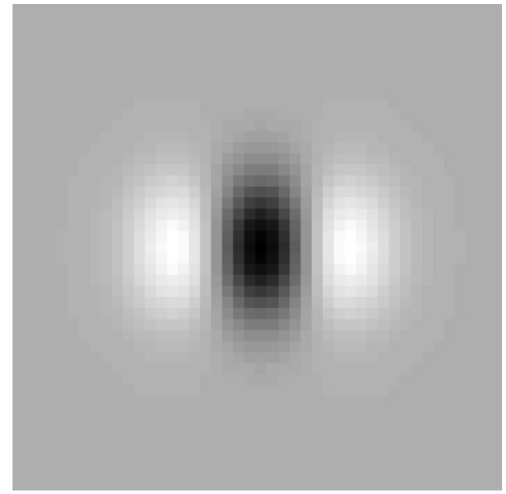
$I(x, y) * \frac{\partial G(x, y, \sigma)}{\partial x}$



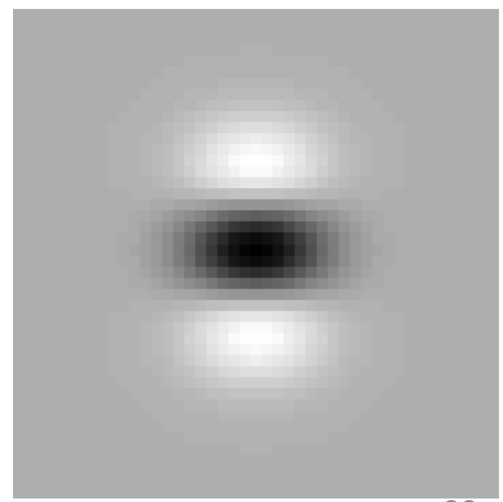
$I(x, y) * \frac{\partial G(x, y, \sigma)}{\partial y}$

Second Derivative of Gaussian

$$\begin{array}{|c|c|c|}\hline 1 & -2 & 1 \\ \hline \end{array} * G(x,y) \approx \frac{\partial^2 G(x,y)}{\partial x^2}$$



$$\begin{array}{|c|}\hline 1 \\ \hline -2 \\ \hline 1 \\ \hline \end{array} * G(x,y) \approx \frac{\partial^2 G(x,y)}{\partial y^2}$$



Laplacian (operator)

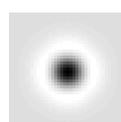
$$\nabla^2 I(x, y) \equiv \frac{\partial^2 I(x, y)}{\partial x^2} + \frac{\partial^2 I(x, y)}{\partial y^2}$$

Laplacian of a Gaussian

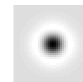
$$\nabla^2 G(x,y,\sigma) \;\equiv\; \frac{\partial^2\, G(x,y,\sigma)}{\partial x^2} + \frac{\partial^2\, G(x,y,\sigma)}{\partial y^2}$$

Not difficult to show that ….

$$= \; -\,\frac{1}{\pi\sigma^4}\left(1 \;-\frac{x^2+y^2}{2\sigma^2}\right)\, e^{-\frac{x^2+y^2}{2\sigma^2}}$$



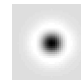
Example: vertical image edge

$I(x, y)$   $\nabla^2 G(x, y, \sigma)$

Q: What happens when we convolve with $\nabla^2 G(x, y, \sigma)$?

Example: vertical image edge

$I(x,y)$ 

 $\nabla^2 G(x,y,\sigma)$

Q: What happens when we convolve with $\nabla^2 G(x,y,\sigma)$?

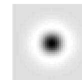
$$\nabla^2 G(x,y,\sigma) \ * I(x,y) \quad \equiv \quad \frac{\partial^2 \ G(x,y,\sigma) * I(x,y)}{\partial x^2} + \boxed{\frac{\partial^2 \ G(x,y,\sigma) * I(x,y)}{\partial y^2}}$$

$= 0$

$I(x,y)$ does not depend on y, and so $G(x,y,\sigma) * I(x,y)$ does not depend on y, and so derivative in y direction is 0.

Example: vertical image edge

$I(x,y)$



$\nabla^2 G(x,y,\sigma)$

Q: What happens when we convolve with $\nabla^2 G(x,y,\sigma)$?

$$\nabla^2 G(x,y,\sigma) \; * I(x,y) \quad \equiv \quad \frac{\partial^2 \; G(x,y,\sigma) * I(x,y)}{\partial x^2}$$

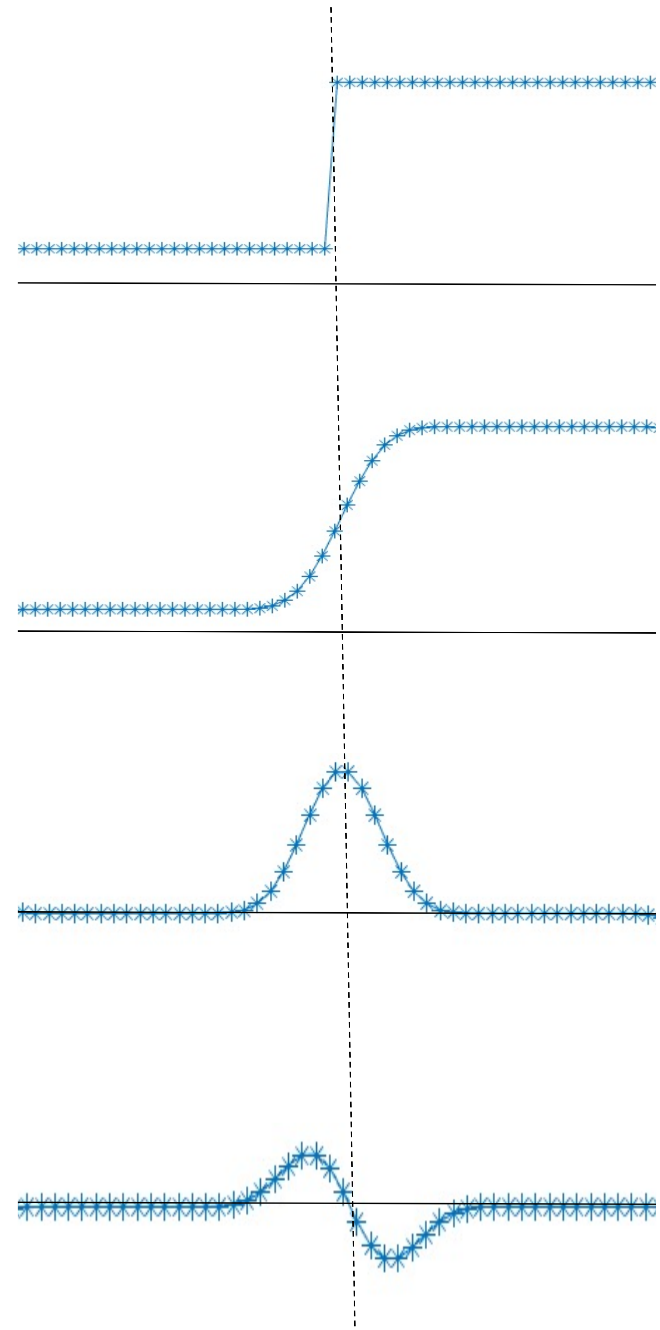
Recall : blurred step edge

$I(x)$

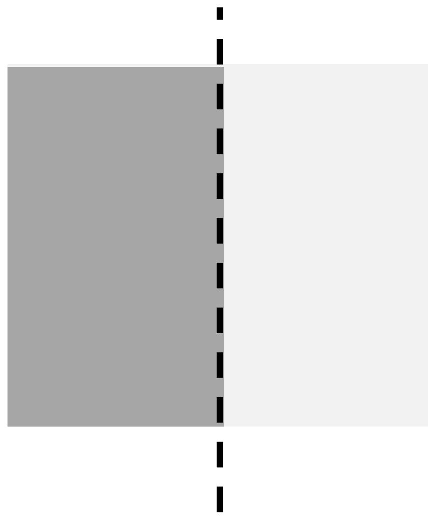
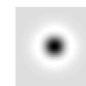
$G(x,\sigma) * I(x)$

$$\frac{d\ G(x,\sigma) *\ I(x)}{dx} \approx$$

$$\boxed{\frac{d^2 G(x,\sigma) * I(x)}{dx^2}} \approx$$



Example: vertical image edge

$I(x, y)$   $\nabla^2 G(x, y, \sigma)$

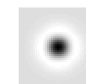
Q: What happens when we convolve with $\nabla^2 G(x, y, \sigma)$?

$$\nabla^2 G(x, y, \sigma) \ * I(x, y) = \frac{\partial^2 \ G(x, y, \sigma) * I(x, y)}{\partial x^2}$$

A: Same as in the 1D case: We get a *zero crossing* at edge.

Example: horizontal image edge

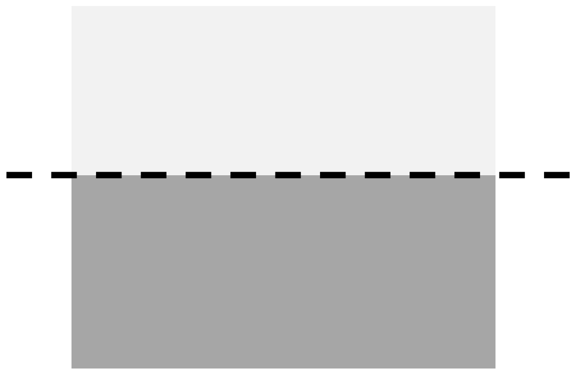
$I(x, y)$ 

$\nabla^2 G(x, y, \sigma)$ 

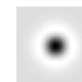
$$\nabla^2 G(x, y, \sigma) * I(x, y) = \quad ?$$

Example: horizontal image edge

$I(x,y)$



$\nabla^2 G(x,y,\sigma)$

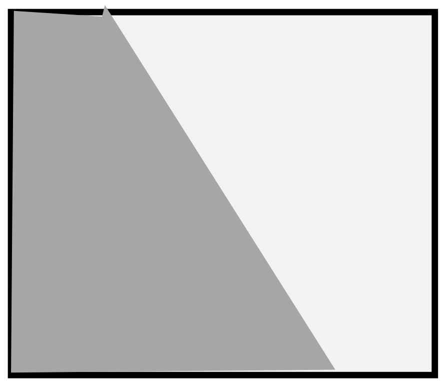


$$\nabla^2 G(x,y,\sigma) \; * I(x,y) = \frac{\partial^2 \, G(x,\sigma)}{\partial y^2} * I(y)$$

Again, we get a *zero crossing* along the edge.

Edge of arbitrary orientation

$I(x, y)$

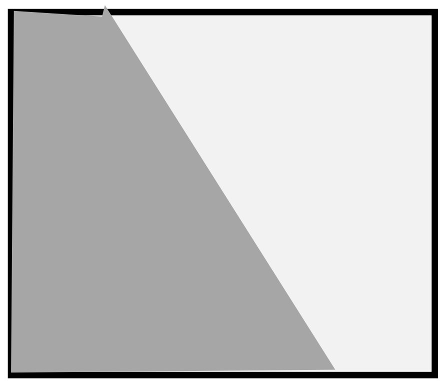


Since $\nabla^2 G(x, y, \sigma)$ is radially symmetric,

$\nabla^2 G(x, y, \sigma) * I(x, y)$ will have a *zero-crossing along an edge of any orientation.*

Marr-Hildreth edge detection (1979)

$I(x, y)$



Compute $\nabla^2 G(x, y, \sigma) * I(x, y)$

The edges are the points (x, y) where there is a zero-crossing.

Marr-Hildreth edge detection (1979)

$$\nabla^2 G(x, y, \sigma) = \Delta\, G(x, y, \sigma)$$

The Laplacian operator Δ is rotationally invariant, in that it is the sum of second partial derivatives in *any two locally orthogonal directions* (for convenience we've used the x and y coordinate axes as these directions).

The theory gets around the problem of having to compute many directional derivatives in 2D.

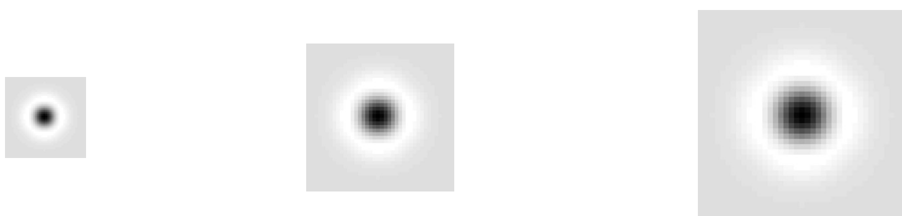
The theoretical justification rests on an assumption of linear variation of intensities in the vicinity of an edge.

[ASIDE: Historical importance of Marr & Hildreth 1979 paper: "Theory of Edge Detection"]

http://rspb.royalsocietypublishing.org/content/207/1167/187

- They showed how to detect edges of arbitrary orientation using a radially symmetric kernel.

- They showed how to use combine the zero-crossings of $\nabla^2 G(x, y, \sigma)$ which are produced using different σ values.



- They argued (convincingly at the time) that this is how the visual system in the human brain works.

ASIDE: What about color ?

- You could detect edges in each of RGB channels and then try to combine these edges.

- You could do edge detection in the HSV channels.