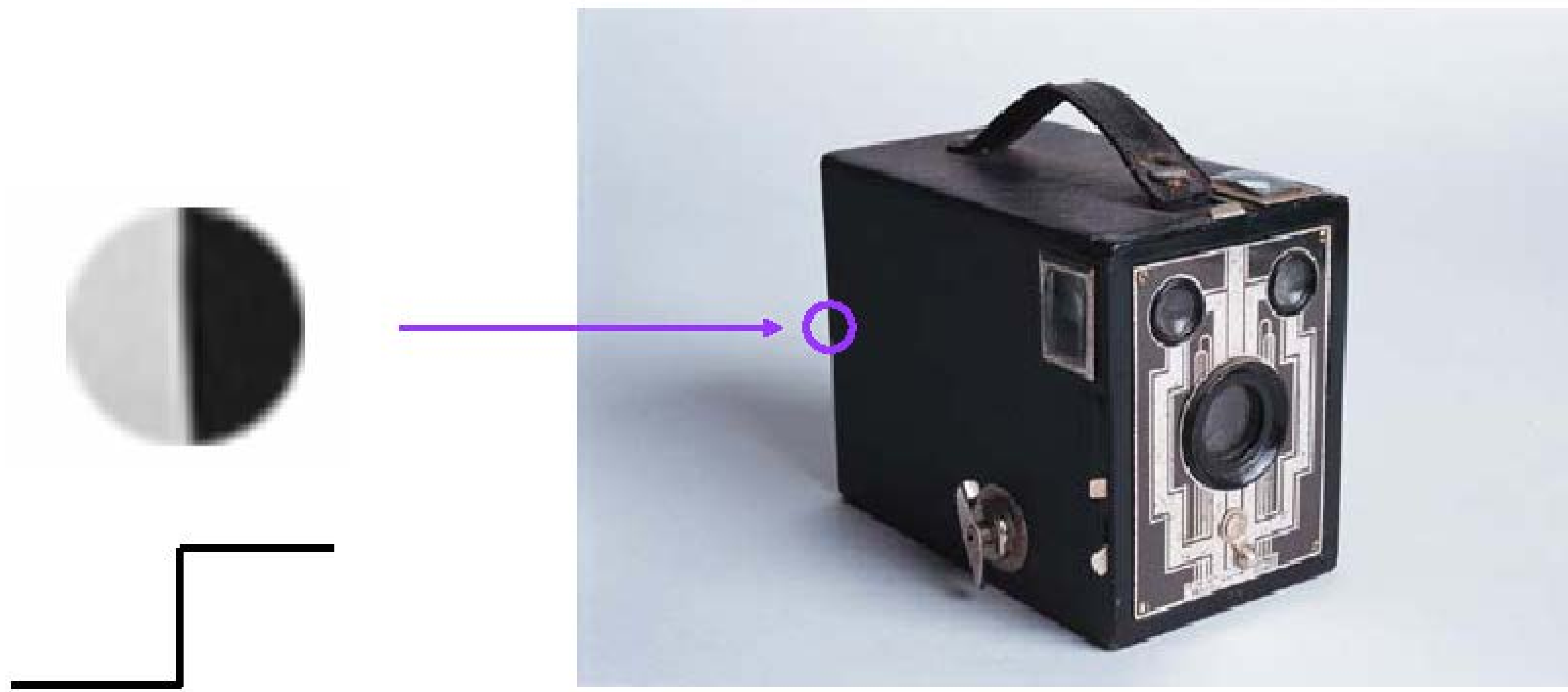
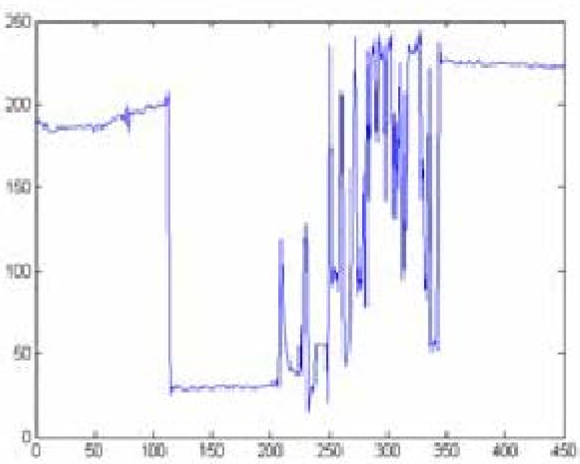
What is a contour ?

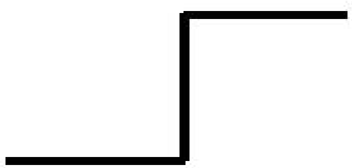
- A contour is a quick variation of intensity.



Source : cis.poly.edu/cs664/

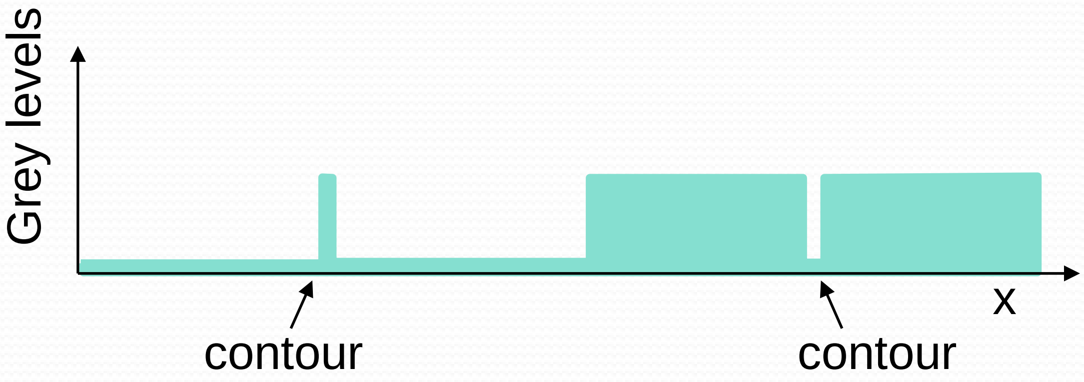
What is a contour ?



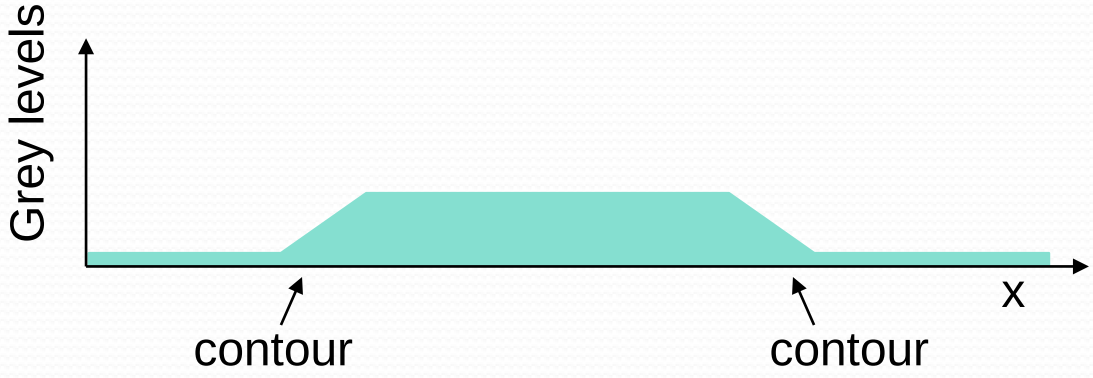


Kind of contours in an image

Step



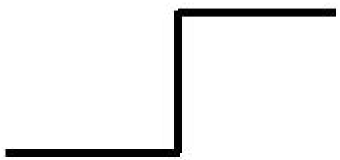
Ramp

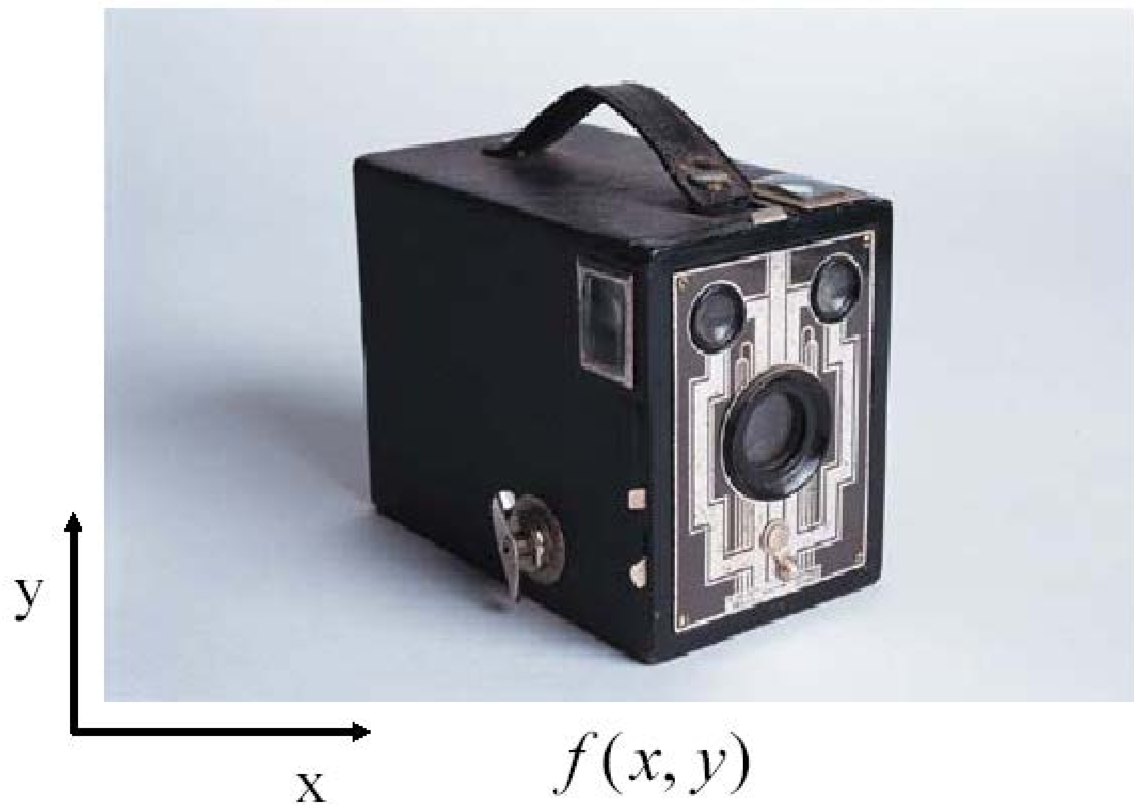


Measure of image variations

- The image gradient (first derivative) is the basic operator to measure the contours in the image.

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





$f(x, y)$

Contour detection

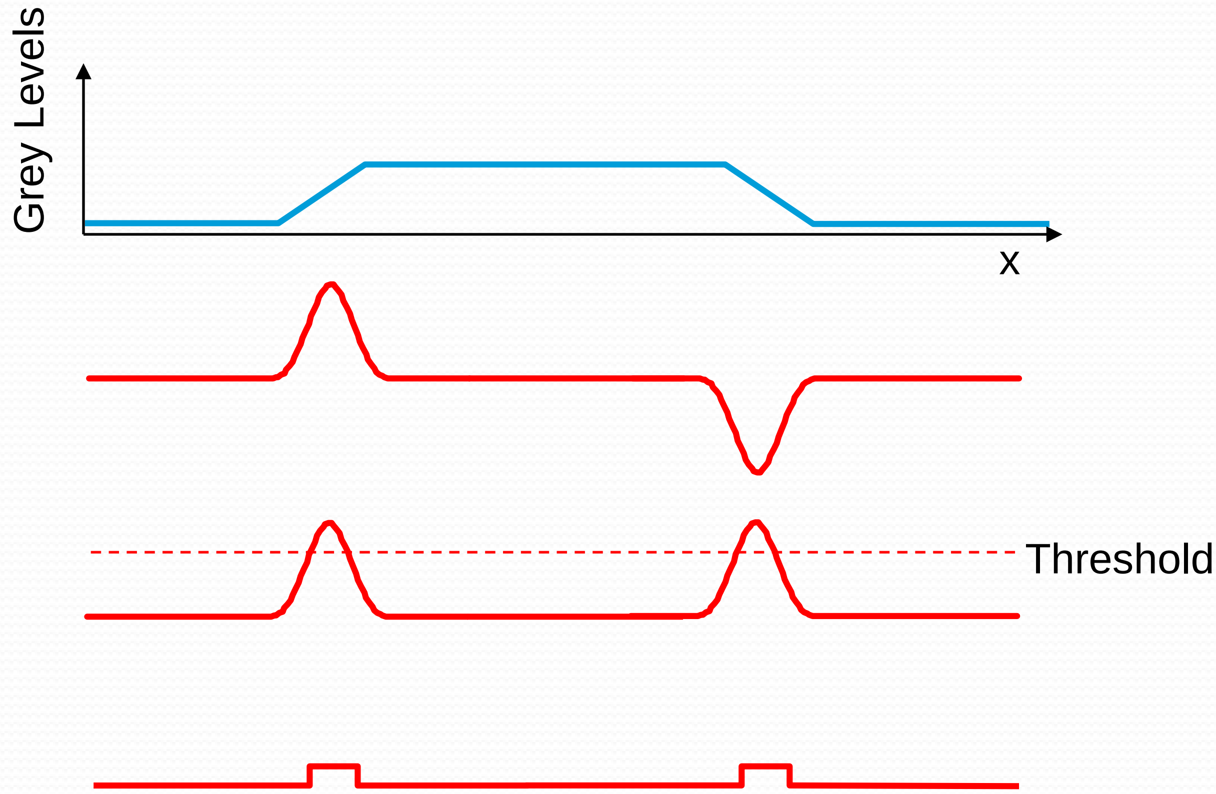
Image 1D f(x)

First derivative f'(x)

|f'(x)|

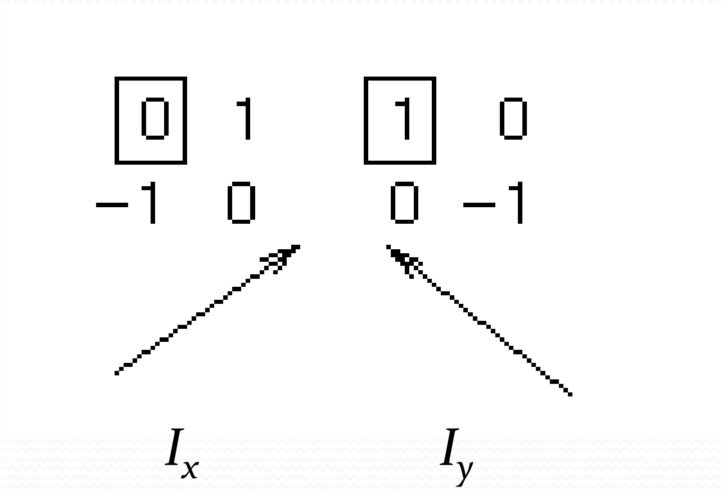
Pixel contours:

|f'(x)| > Threshold



First image derivative (Roberts)

- Roberts (1965) provides a first approximation of the first derivative of a discrete image.
- The calculation is done with two masks of convolution representing the derivative in two orthogonal directions.



First image derivative (Roberts)

- The norm of the gradient (mainly used) is obtained by:

$$|\nabla I| = \sqrt{{I_x}^2 + {I_y}^2} \approx |I_x| + |I_y|$$

- The direction of the gradient (less used) is obtained by:

$$\theta = \arctan(I_y / I_x) - 3\pi / 4$$

Edge detection operators

- Subsequently, several other approximations of the discrete gradient appeared.
- Examples:
  - Prewitt

$$Px = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \text{ et } Py = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

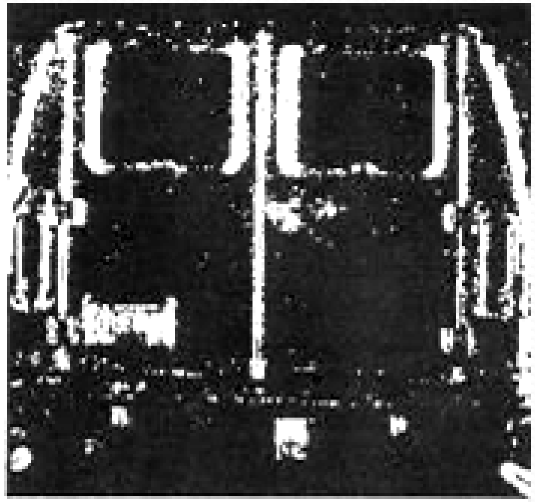
  - Sobel

$$Sx = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \text{ et } Sy = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

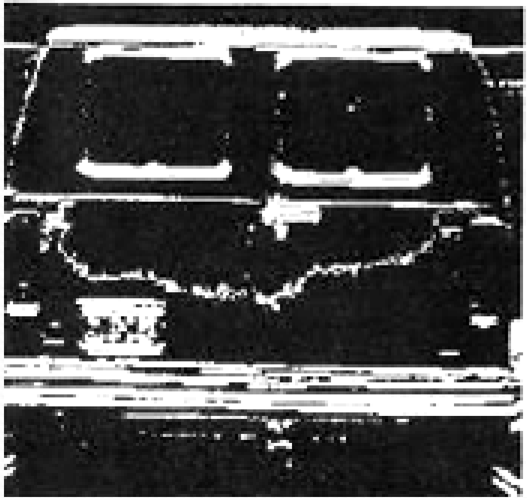
  - Kirsh: 8 masks for 8 directions
  - Canny, Deriche: optimisation of vriatianal criteria

Example of edge detection

Image $\boldsymbol{I}$

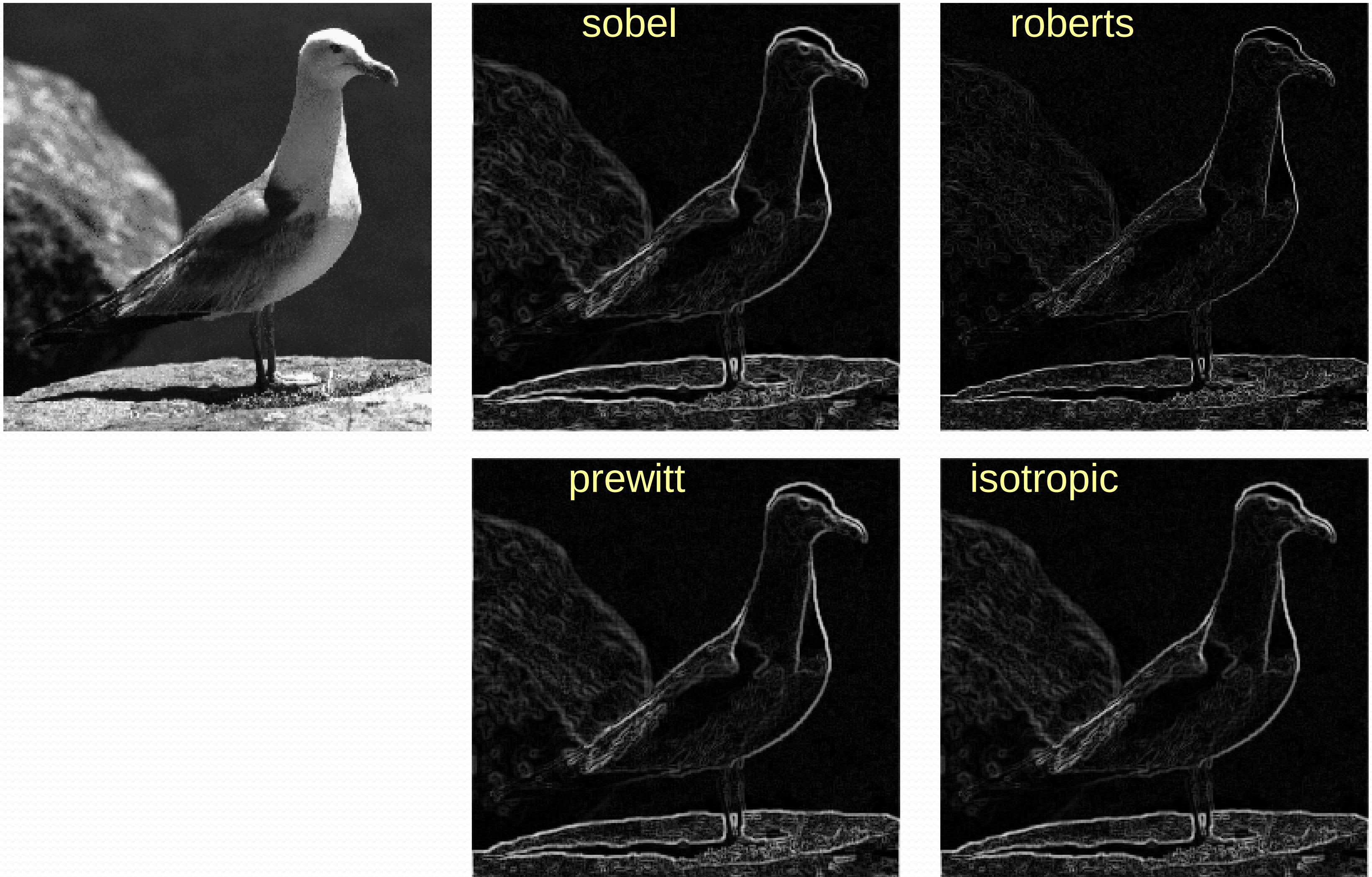


Vertical Edges $\dfrac{\partial I}{\partial x}$



Horizontal Edges $\dfrac{\partial I}{\partial y}$

Example of edge detectors

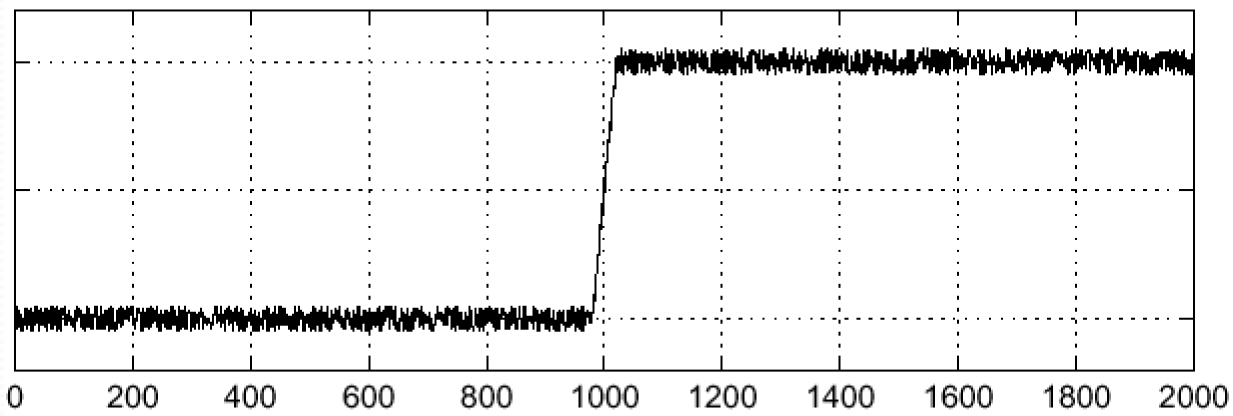


Edge point detectors

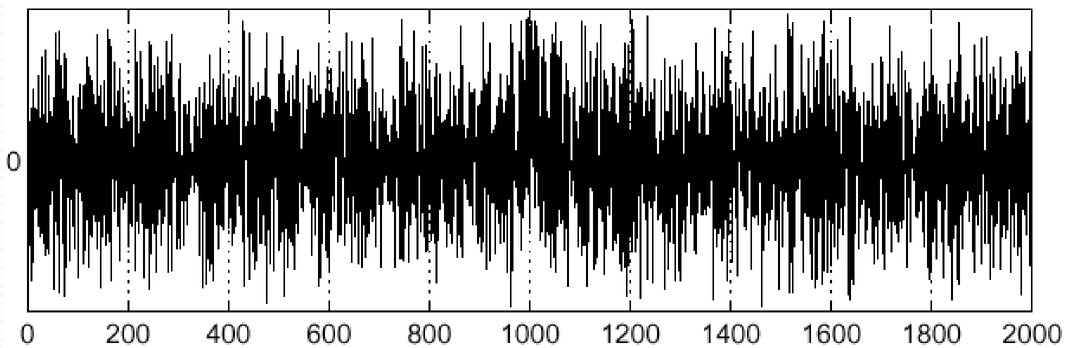
- There are several operators to detect points of contours (derived from the first order).
- Most of these operators (except Roberts) smooth the image in order to obtain a better result.
- In practice, one obtains the incomplete edges
  - There are extra pixels
  - There are missing pixels
  - there are errors in the position and orientation of the edge pixels
- We will have to consider other techniques to complete the indices of contours obtained with these operators.

Effect of smoothing on derivative

$f(x)$



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



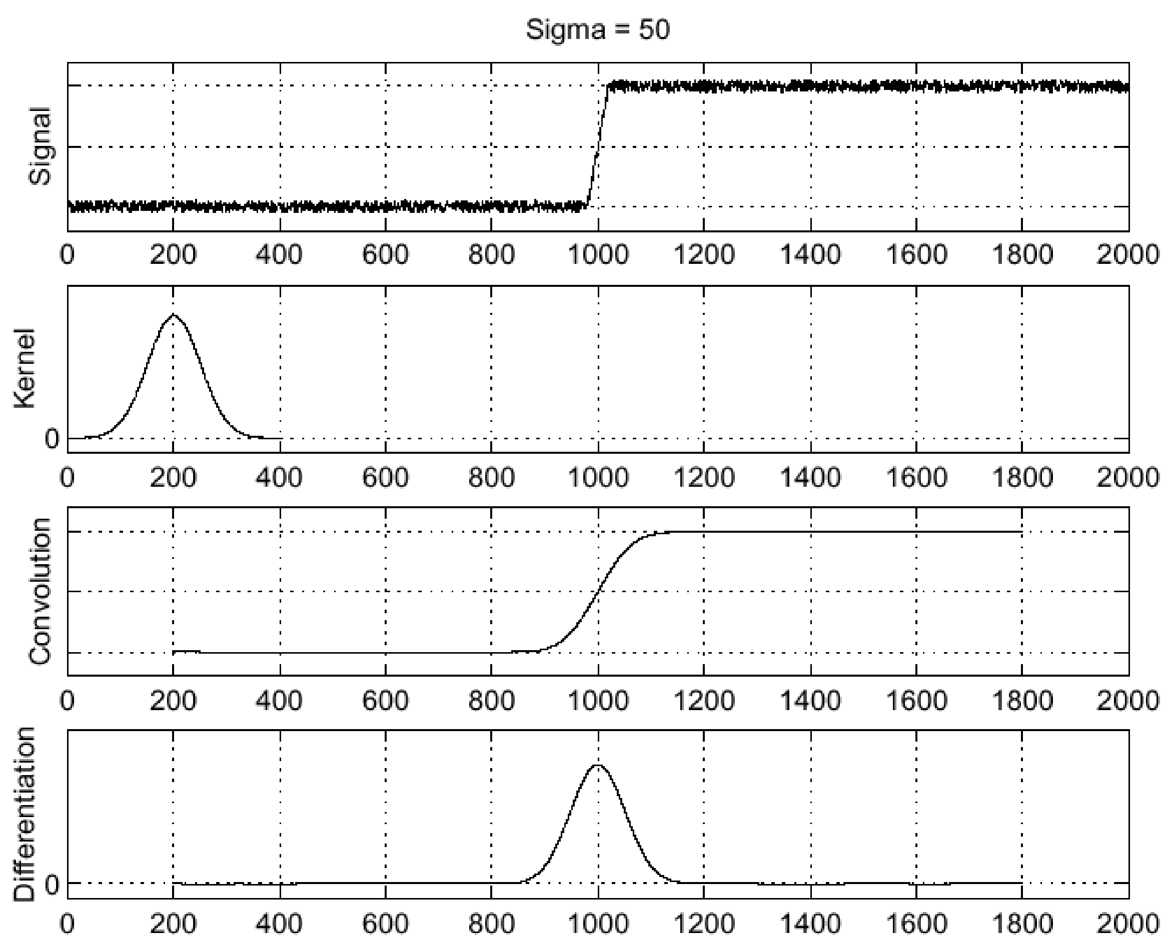
Effect of smoothing on derivative

f

h

$h \star f$

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



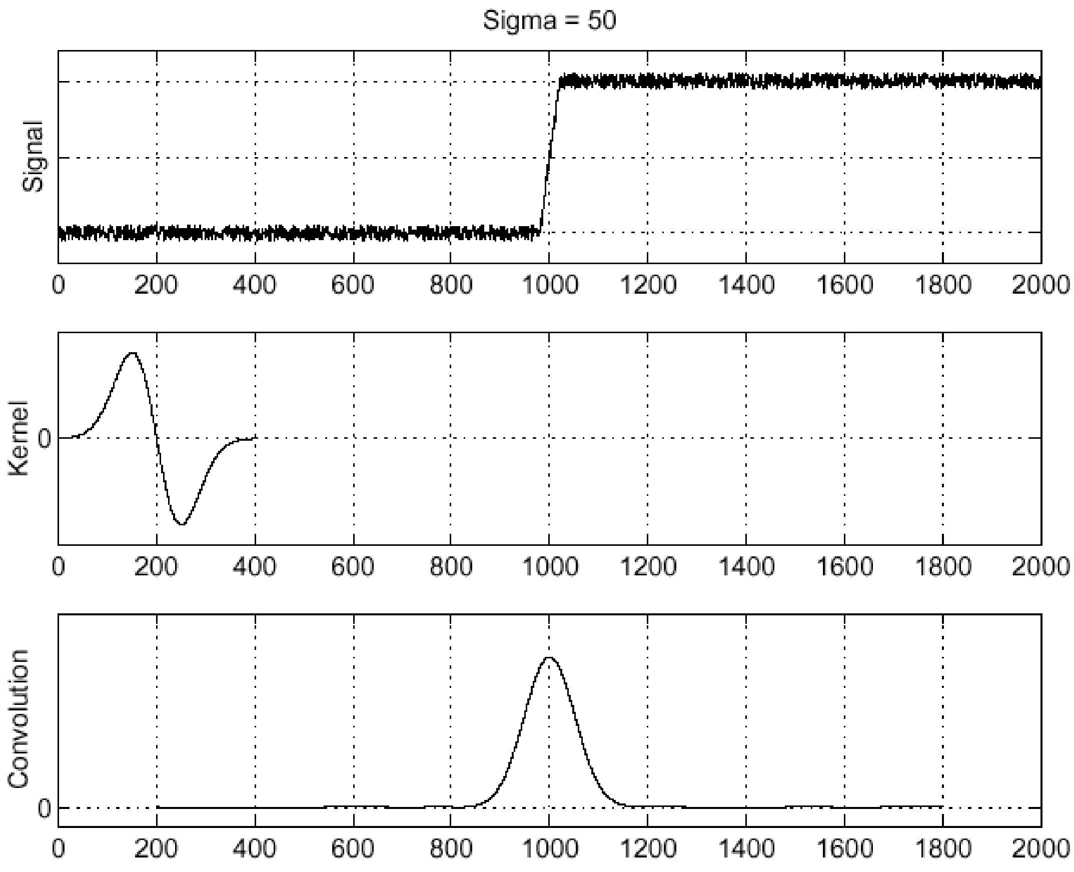
Combining smoothing and derivation

$$\frac{\partial}{\partial x}(h \star f) = (\frac{\partial}{\partial x}h) \star f$$ (i.e., saves one step)

f

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

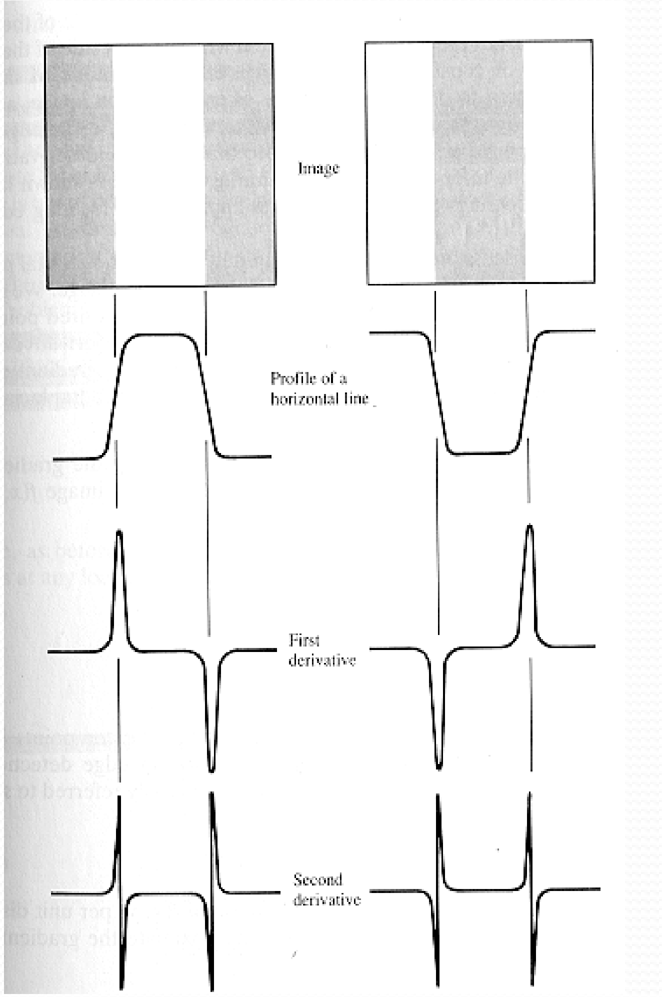
$(\frac{\partial}{\partial x}h) \star f$



Second derivative of the image

- Another approach to find the edges of the image is to use the second derivative of the image.
- To do this, we use the Laplacian as operator
- contours correspond:
  - At maxima of the first derivative
  - At vanishing points of the second derivative

$$\nabla^2 I = \frac{\partial I}{\partial x^2} + \frac{\partial I}{\partial y^2}$$



Laplacian

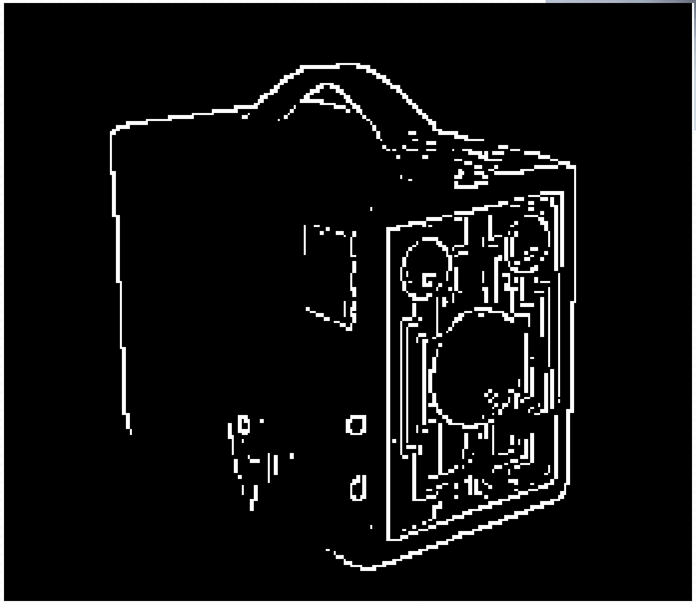
- Several discrete approximations of the Laplacian exist.

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{bmatrix} \quad \text{or} \ldots$$

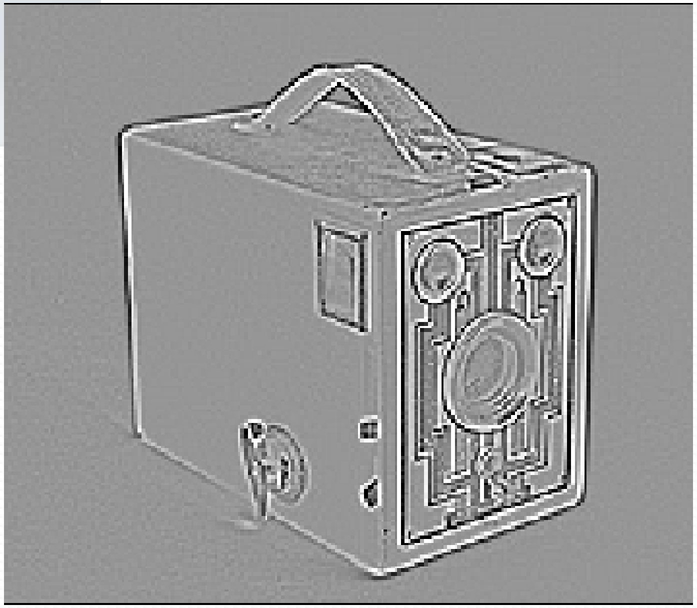
- A single matrix convolution
- A matrix symetric by rotation

Comparison Gradient / Laplacian

Gradient



Laplacian



Laplacian of Gaussian

- The Laplacian, like all contour detectors, is very sensitive to noise.
- We thus prefer smoothing the image before detecting the contours.

- To do this, we combine a Gaussian smoothing with the Laplacian.
  - The Gaussian smooth the image and makes the contours blurred, but preserves their positions
  - The Laplacian determines contours as vanishing positions of the second derivative

Laplacian of Gaussian

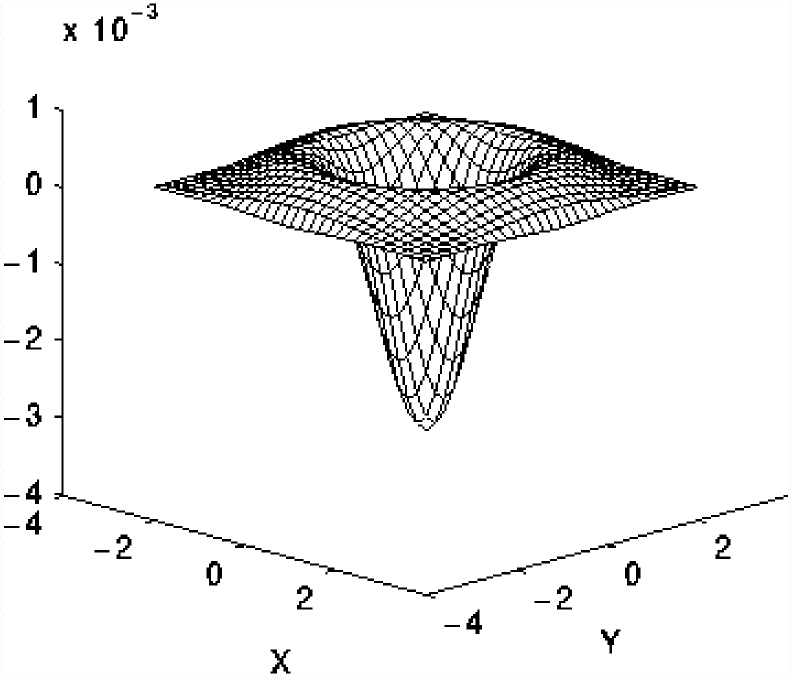
- We make this image operation in one step by using a filter of Laplacian of Gaussian *(LoG) :*

$$LoG * I = \nabla^2 * G * I = \nabla^2 G * I$$

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

- In this filter, we first convolve the Laplacian and the Gaussian by choosing the parameter σ of the Gaussian.
- The human visual system would use a similar approach to recognize contours.

Laplacian of Gaussian



The LoG function has the form of a reverse Mexican hat.

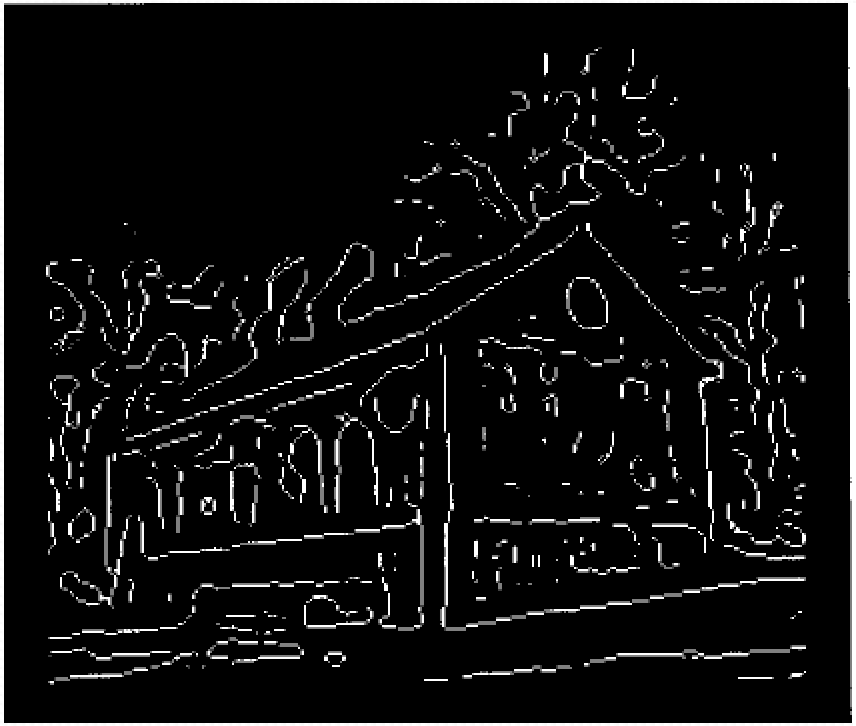
$$
\begin{bmatrix}
0 & 1 & 1 & 2 & 2 & 2 & 1 & 1 & 0 \\
1 & 2 & 4 & 5 & 5 & 5 & 4 & 2 & 1 \\
1 & 4 & 5 & 3 & 0 & 3 & 5 & 4 & 1 \\
2 & 5 & 3 & -12 & -24 & -12 & 3 & 5 & 2 \\
2 & 5 & 0 & -24 & -40 & -24 & 0 & 5 & 2 \\
2 & 5 & 3 & -12 & -24 & -12 & 3 & 5 & 2 \\
1 & 4 & 5 & 3 & 0 & 3 & 5 & 4 & 1 \\
1 & 2 & 4 & 5 & 5 & 5 & 4 & 2 & 1 \\
0 & 1 & 1 & 2 & 2 & 2 & 1 & 1 & 0
\end{bmatrix}
$$

Discrete approximation of LoG with σ=1.4

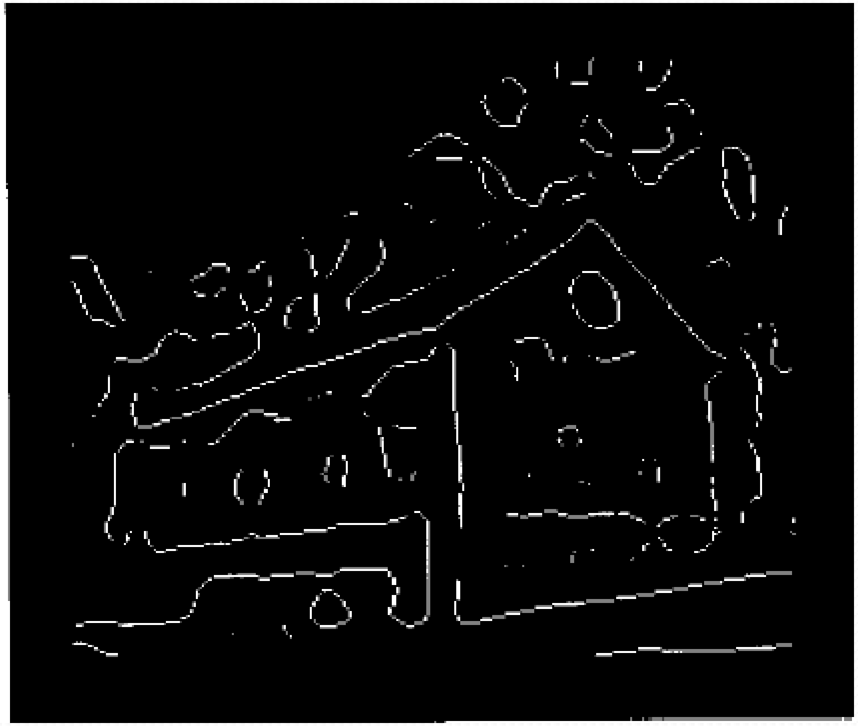
We often choose 5 σ for the size of the gaussian filter.*

Source: http://www.dai.ed.ac.uk/HIPR2/log.htm

Laplacian of Gaussian



Sigma = 3



Sigma = 5

By varying σ, we get different levels of contours.

Which filter to choose for detecting edges ?

- No operator is perfect for detecting edges.
- In practice, one obtains the incomplete edges
  - There are extra pixels
  - There are missing pixels
  - there are errors in the position and orientation of the edge pixels
- Each one seems to have a preference for a method or another.
- A contour detector is only a first step in the chain of image segmentation.