University of Stavanger

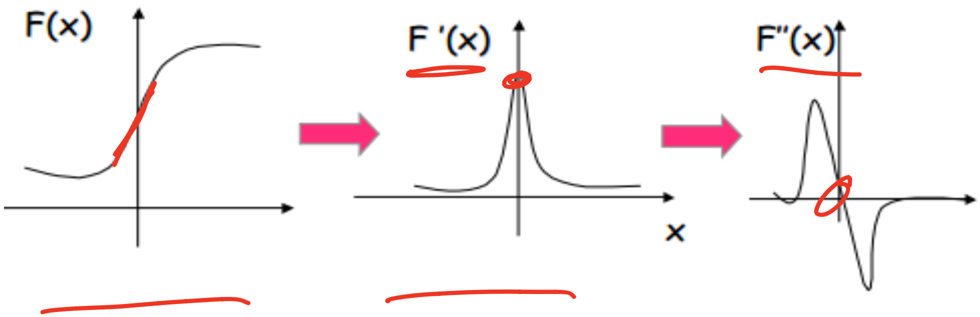
Prof. Kjersti Engan

ELE510 Image processing and computer vision

Spatial-Domain Filtering, second derivative (chap 5.4 Birchfield) 2020

- Recall Sharp changes in gray level of the input image correspond to "peaks or valleys" of the first-derivative of the input signal.

- Peaks or valleys of the first-derivative of the input signal, correspond to "zero-crossings" of the second-derivative of the input signal.



(5.4) Second derivative

- Finding the second derivative can be seen in 1D by convolving the function with the noncentralized difference operator twice:

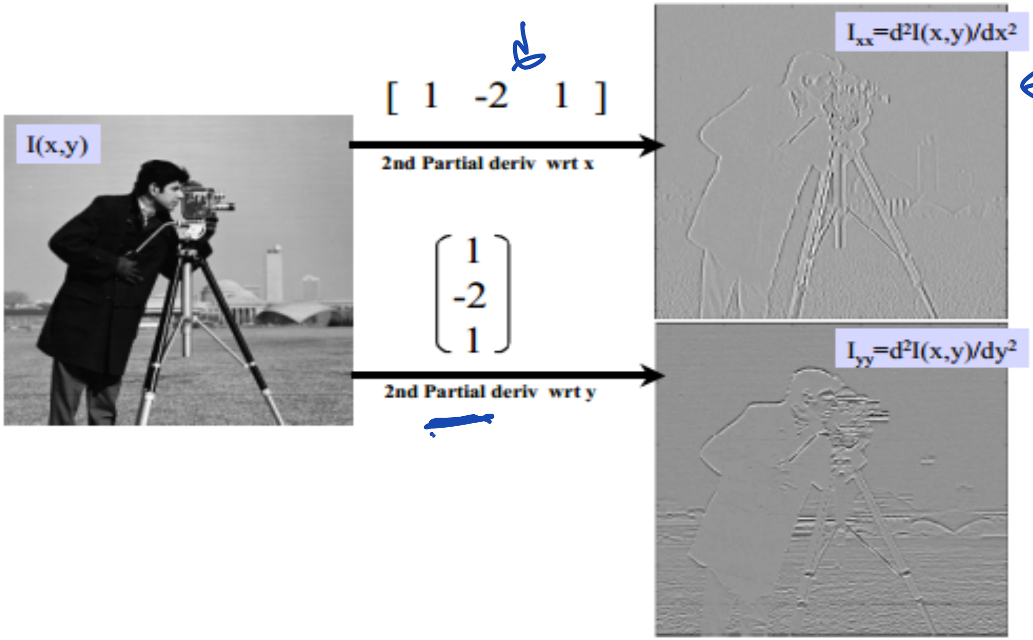
$$(f(x) \circledast [1 \quad -1]) \circledast [1 \quad -1] = f(x) \circledast ([1 \quad -1] \circledast [1 \quad -1]) = f(x) \circledast [1 \quad -2 \quad 1]$$

- In 2D, the second-derivative in the *x* and *y* directions can be obtained by convolving with the second-derivative kernel:

$$\frac{\partial^2 I(x, y)}{\partial x^2} = I(x, y) \circledast [1 \quad -2 \quad 1] \, .$$

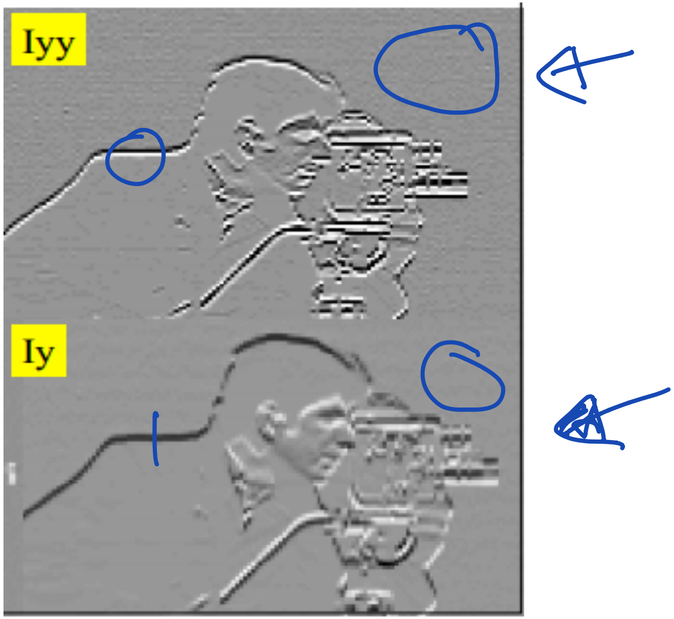
$$\frac{\partial^2 I(x, y)}{\partial y^2} = I(x, y) \circledast \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} \, .$$

Second derivatives and convolution



Second derivatives and convolution

- Better localized edges
- But more sensitive to noise



Laplacian filter

- Taking the second derivative of a function (image) gives a zero-crossing at and edge.

- The Laplacian of an image f can be find as:

$$\Delta f = \nabla \cdot \nabla f = \nabla^2 f$$

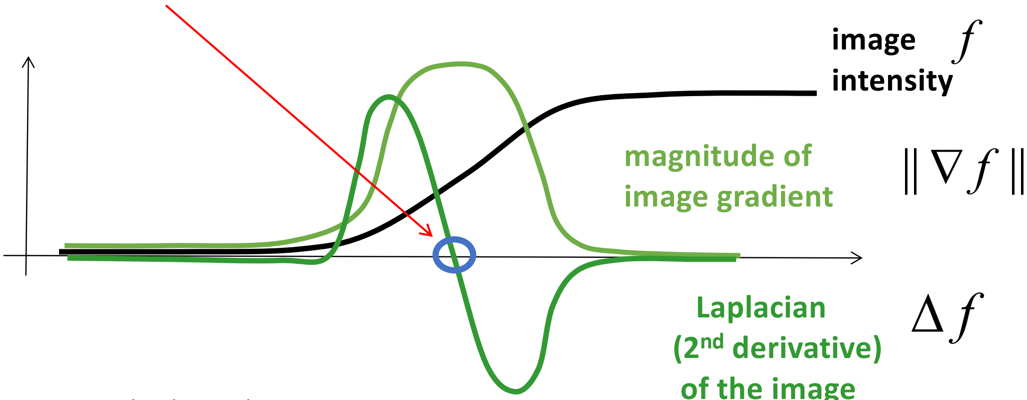
$$\nabla^2 f(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}$$

Second Image Derivatives

- Laplacian Zero Crossing

$$\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

- Used for edge detection
  (alternative to computing Gradient extrema)



http://homepages.inf.ed.ac.uk/rbf/HIPR2/zeros.htm

Laplacian filtering

- Laplace operator – approximation filter mask

rotationally invariant
second derivative for 2D functions:

$$
\begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix}
+
\begin{bmatrix} 0 & -1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 0 \end{bmatrix}
=
\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}
$$

Finite Difference
Second Order
Derivative in x

Finite Difference
Second Order
Derivative in y

But problems with noisy images.

$$[1 \;\; -2 \;\; 1] = I_{xx}$$

$$\begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} = I_{yy}$$

Laplacian filtering

h

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} * I$$

- -Zero on uniform regions
- -Positive on one side of an edge
- -Negative on the other side
- -Zero at some point in between on the edge itself

➔ band-pass filter (Suppresses both high and low frequencies)

http://www.cse.psu.edu/~rcollins/CSE486/lecture11_6pp.pdf

Laplacian – some remarks

- Can be found using a single mask (1.st derivative needs two)
- Orientation is lost
- Taking derivatives increase noise
  - Second derivative is very noise sensitive!
- Should be combined with a smoothing ....

  if we first do gaussion smoothing -> LoG filter

$$\boxed{\begin{matrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{matrix}}$$

Laplacian of Gaussian

Laplacian of Gaussian (LoG)

- **Laplacian operator ∇^2**: defined as the divergence of the gradient of a function.

$$\Delta I : \nabla^2 I = \nabla \cdot \nabla I = \begin{bmatrix} \frac{\partial}{\partial x} & \frac{\partial}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial I}{\partial x} & \frac{\partial I}{\partial y} \end{bmatrix}^{\mathsf{T}} = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

- Computing the Laplacian of a smoothed image is the same as convolving the image with the **Laplacian of Gaussian (LoG):**

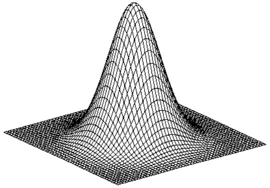
$$\frac{\partial^2 (I \circledast Gauss(x,y))}{\partial x^2} + \frac{\partial^2 (I \circledast Gauss(x,y))}{\partial y^2} = I \circledast \underbrace{\left(\frac{\partial^2\, Gauss(x,y)}{\partial x^2} + \frac{\partial^2\, Gauss(x,y)}{\partial y^2} \right)}_{LoG}$$

- LoG also called Mexican hat
- 2D Gaussian second derivative is separable

$$I(x,y) \circledast \overbrace{\ddot{g}auss(x) \circledast gauss(y)} +$$

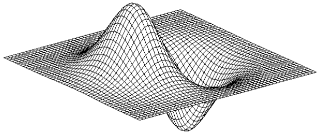
$$I(x,y) \circledast \ddot{g}auss(y) \circledast gauss(x)$$

Laplacian of Gaussian (LoG)



Gaussian

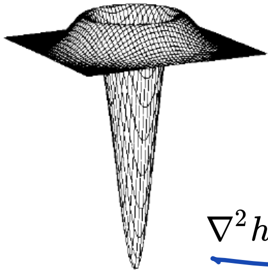
$$h_\sigma(u,v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$



derivative of Gaussian

$$\frac{\partial}{\partial x} h_\sigma(u,v)$$

Laplacian of Gaussian



$$\nabla^2 h_\sigma(u,v)$$

∇^2 is the **Laplacian** operator:

$$\nabla^2 f(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}$$

Laplacian of Gaussian (LoG)

- Gaussian function: $$G_0(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Disregard scaling: $$G(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Laplacian of function f(x,y):

$$\Delta f(x,y) = \nabla^2 f(x,y) = \frac{d^2 f(x,y)}{dx^2} + \frac{d^2 f(x,y)}{dy^2}$$

- LoG:

$$\boxed{\nabla^2 G(x,y) = \frac{x^2+y^2-2\sigma^2}{\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}}}$$

σ

Laplacian of Gaussian (LoG) 3x3 kernels

The only possible for 3x3 Gaussian 1. order differenting kernel is ½ [1 0 -1]. The corresponding only 2.order gauss diff. 3x3 kernel is [1 -2 1]
However, different smoothing functions (Gaussian with different varians) in the LoG gives different LoG kernels:

$\sigma^2 = 0.0$	$\sigma^2 = 0.167$	$\sigma^2 = 0.20$	$\sigma^2 = 0.25$	$\sigma^2 = 0.33$	$\sigma^2 = 0.5$
$[0 \quad 1 \quad 0]$	$\frac{1}{12}[1 \quad 10 \quad 1]$	$\frac{1}{10}[1 \quad 8 \quad 1]$	$\frac{1}{8}[1 \quad 6 \quad 1]$	$\frac{1}{6}[1 \quad 4 \quad 1]$	$\frac{1}{4}[1 \quad 2 \quad 1]$
$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	$\frac{1}{6}\begin{bmatrix} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{bmatrix}$	$\frac{1}{5}\begin{bmatrix} 1 & 3 & 1 \\ 3 & -16 & 3 \\ 1 & 3 & 1 \end{bmatrix}$	$\frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \\ 2 & -12 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	$\frac{1}{3}\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	$\frac{1}{2}\begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & 0 & 1 \end{bmatrix}$

TABLE 5.5 Various discrete 3 × 3 LoG kernels. For each choice of variance, the middle row shows the 1D smoothing kernel, while the last row shows the resulting LoG kernel.

3x3 double diff kernel : [1 -2 1]

smoothing can vary (σ)

$\sigma^2 = 0.25 \rightarrow \frac{1}{8}\begin{bmatrix}1\\6\\1\end{bmatrix}$

$$\frac{\partial^2 Gauss_{0.25}}{\partial x^2} = [1 \quad -2 \quad 1] \circledast \frac{1}{8}\begin{bmatrix}1\\6\\1\end{bmatrix}$$

$$= \frac{1}{8}\begin{bmatrix}1 & -2 & 1\\6 & -12 & 6\\1 & -2 & 1\end{bmatrix}$$

$$\frac{\partial^2 gauss_{0.25}}{\partial y^2} = \frac{1}{8}[1 \quad 6 \quad 1]\begin{bmatrix}1\\-2\\1\end{bmatrix} \Rightarrow \frac{1}{8}\begin{bmatrix}1 & 6 & 1\\-2 & -12 & -2\\1 & 6 & 1\end{bmatrix}$$

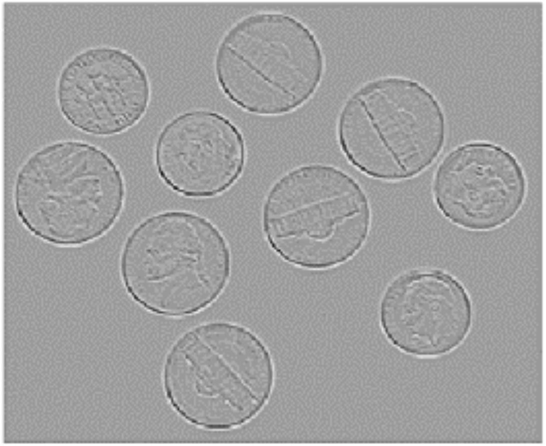
$$LoG_{0.25} = \frac{\partial^2}{\partial x^2} g_{0.25} + \frac{\partial^2}{\partial y^2} g_{0.25}$$

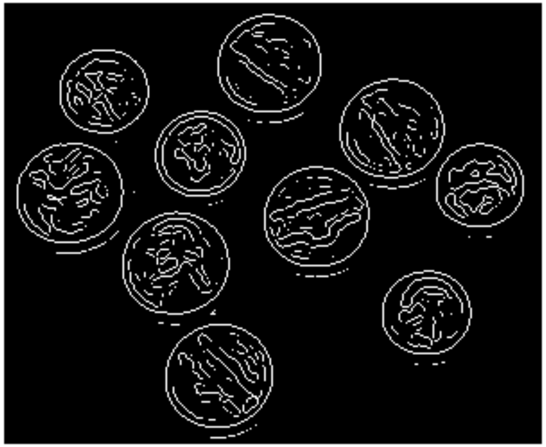
$$= \frac{1}{8}\begin{bmatrix}1 & -2 & 1\\6 & -12 & 6\\1 & -2 & 1\end{bmatrix} + \frac{1}{8}\begin{bmatrix}1 & 6 & 1\\-2 & -12 & -2\\1 & 6 & 1\end{bmatrix}$$

$$= \frac{1}{4}\boxed{\begin{bmatrix}1 & 2 & 1\\2 & -12 & 2\\1 & 2 & 1\end{bmatrix}}$$

LoG as edge detector

LoG gives doble edge image.   Can use zero-crossings to find edges.





Matlab: edge(Im,'log',T,sigma)

T – threshold on edges. sigma – σ of Gaussian filter

LoG – closed contours

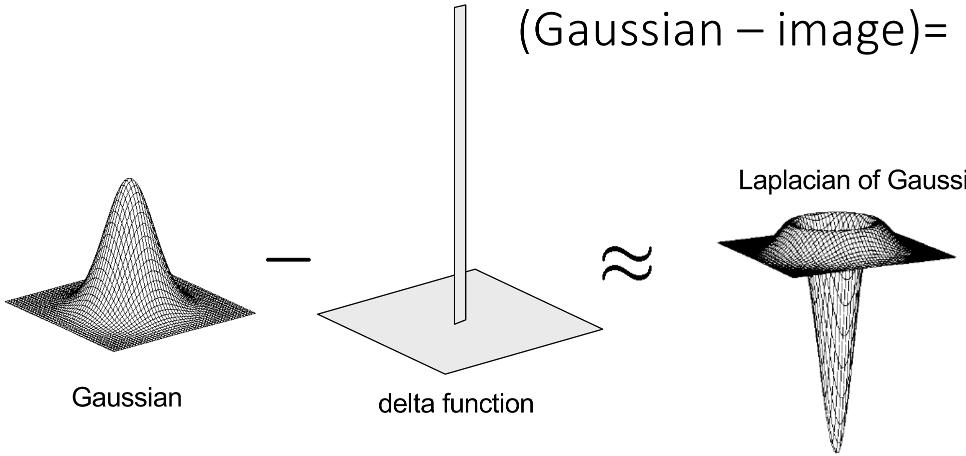
M=10

M=20

If edge map defined from zero-crossings at LoG output, we get closed contours.

Difference of Gaussian (DoG)

(Gaussian – image)= DoG $\approx$ LoG of image



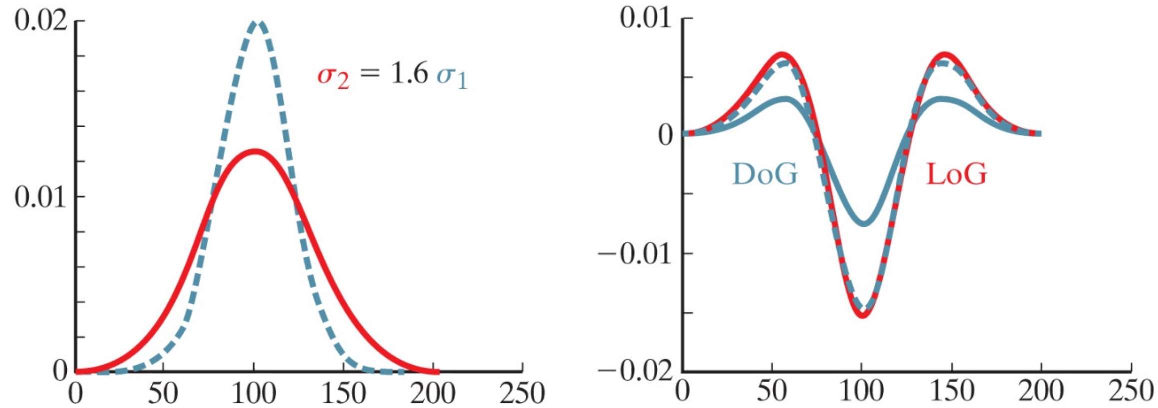
Edge detection by subtraction (DoG)

original

smoothed (5x5 Gaussian)

smoothed – original
(scaled by 4, offset +128)

Figure 5.15 LEFT: Two Gaussians whose ratio of standard deviations is 1.6. RIGHT: The difference of Gaussians (solid blue) and 1D Laplacian of Gaussian (solid red). The scaled DoG (dashed blue) approximates the LoG.



LoG can be approximated by a difference between two Gaussians at different scales

Best approx. when $\sigma_1 = \dfrac{\sigma}{\sqrt{2}},\ \sigma_2 = \sqrt{2}\sigma$