Marr-Hildreth edge detection

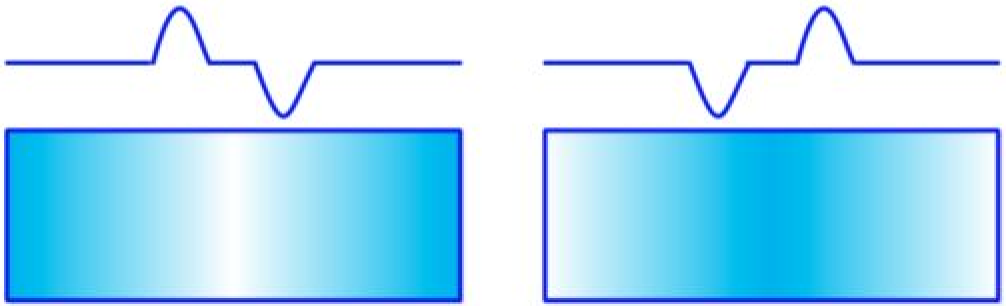
Xuan Yang and Chao Duan

Overview

- Basic idea of edge detection
- Second order & Laplacian
- LoG operator(Marr-Hildreth operator)

Basic Idea of edge detection

First order differentioation



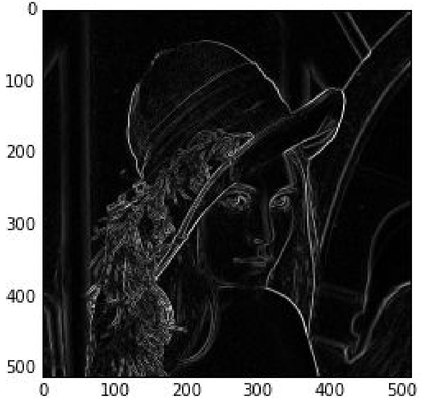
For the vertical edges Ex:

$$\mathbf{Ex}_{x,y} = \left|\mathbf{P}_{x,y} - \mathbf{P}_{x+1,y}\right| \qquad \forall x \in 1, N-1;\ y \in 1, N$$

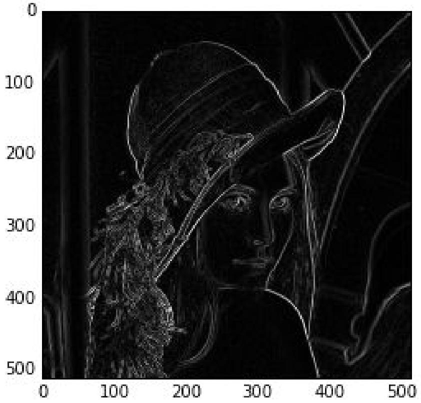
$$\mathbf{Ey}_{x,y} = \left|\mathbf{P}_{x,y} - \mathbf{P}_{x,y+1}\right| \qquad \forall x \in 1, N;\ y \in 1, N-1$$

Ref.: Edge detection, http://blog.csdn.net/xiaowei_cqu/article/details/7829481

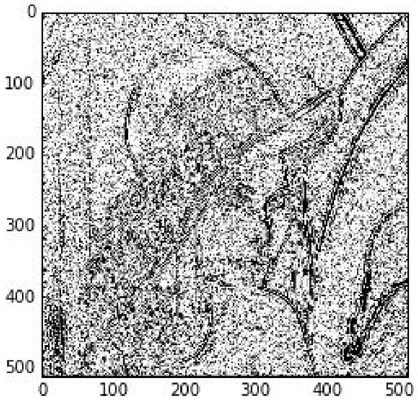
First order vs second order



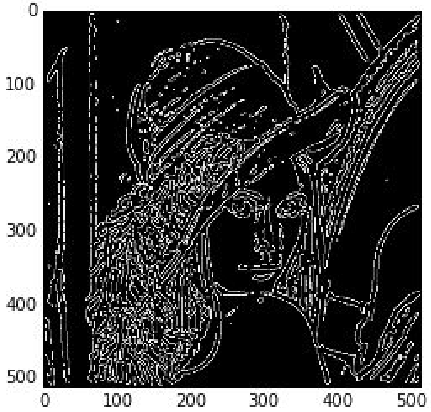
prewitt



sobel

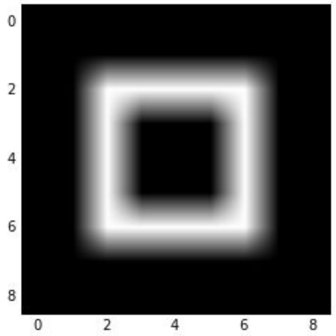


Laplacian without smoothed



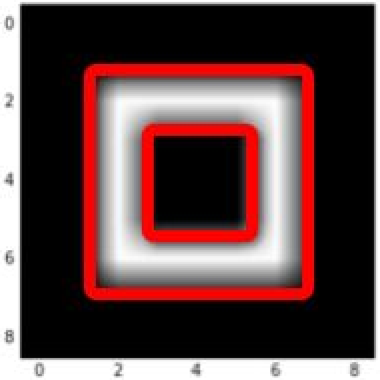
Marr-H with 15x15 template
sigma = 2.3

First order vs second order



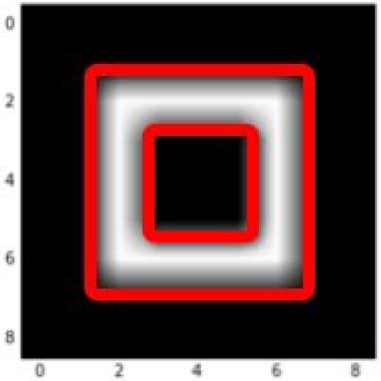
Original

First order vs second order

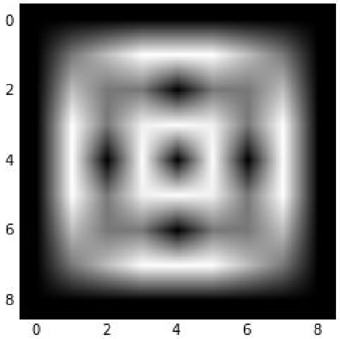


Original

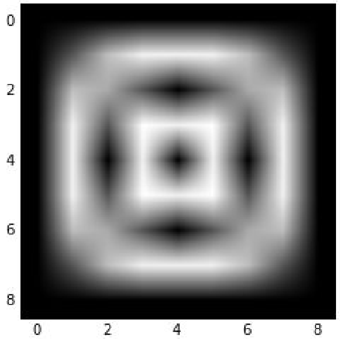
First order vs second order



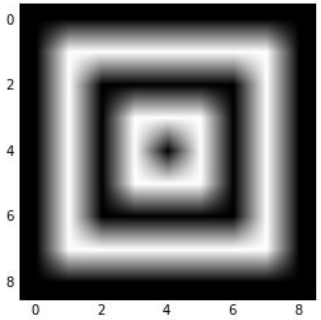
Original



Prewitt

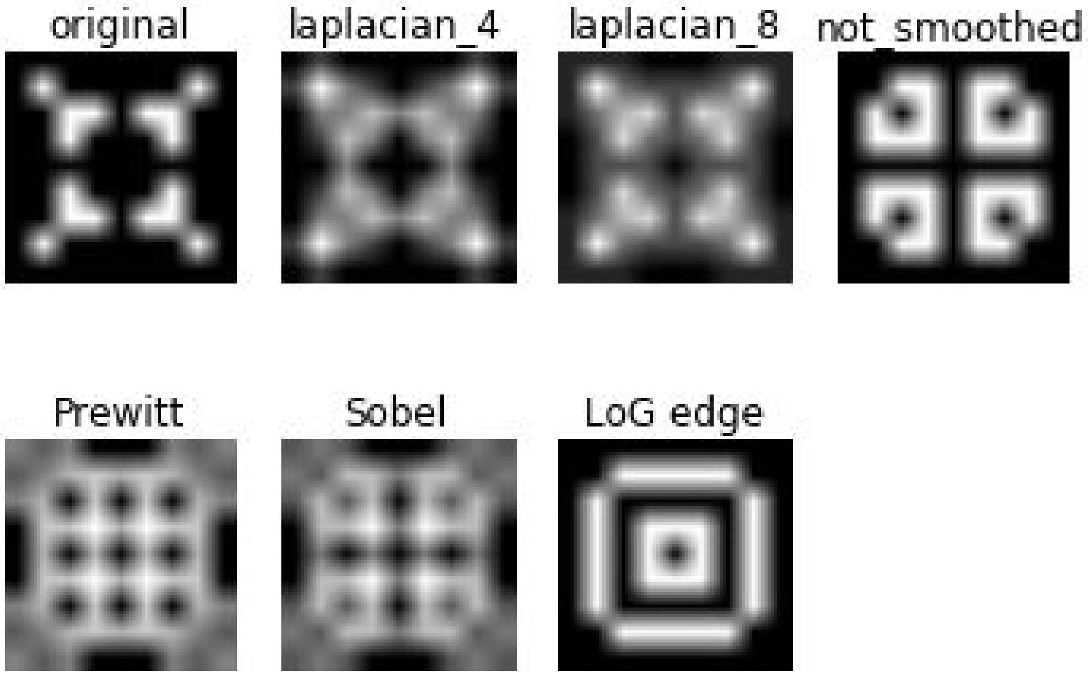


Sobel



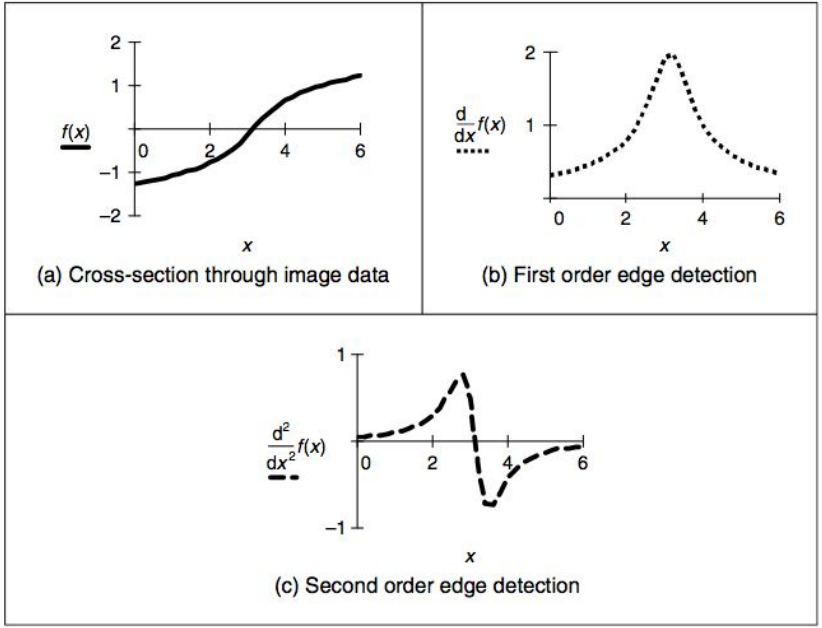
Laplacian operator

First order vs second order



The second order operator—the Laplacian

The Laplacian operator



first and second edge detection from [1]

Ref.: Nixon M. Feature extraction & image processing[M]. Academic Press, 2008.

The Laplacian operator

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

The Laplacian operator

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

$$\begin{aligned}\nabla^2 f(x, y) = &\{[f(x+1, y) - f(x, y)] - [f(x, y) - f(x-1, y)]\} \\ &+ \{[f(x, y+1) - f(x, y)] - [f(x, y) - f(x, y-1)]\}\end{aligned}$$

The Laplacian operator

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

$$\begin{aligned}\nabla^2 f(x, y) = &\{[f(x+1, y) - f(x, y)] - [f(x, y) - f(x-1, y)]\} \\ &+ \{[f(x, y+1) - f(x, y)] - [f(x, y) - f(x, y-1)]\}\end{aligned}$$

$$= f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4 f(x+1, y)$$

The Laplacian operator

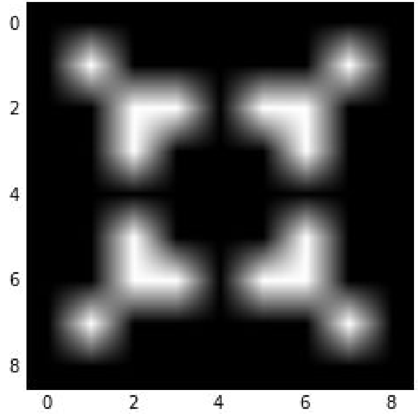
$$\nabla^2 f(x,y) = f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1) - 4f(x,y)$$

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

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

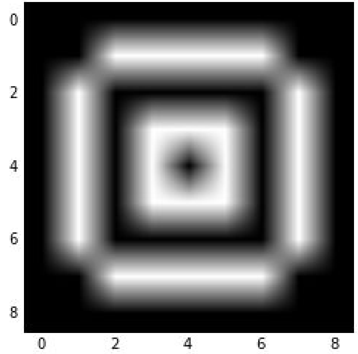
The Laplacian operator

Original

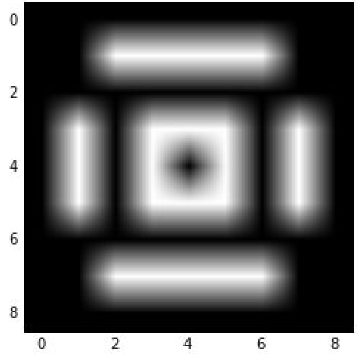


(Different template with zero crossing)

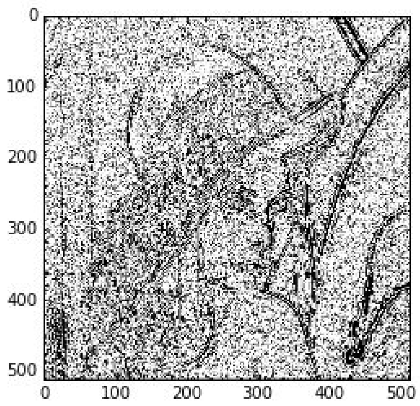
$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$



$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

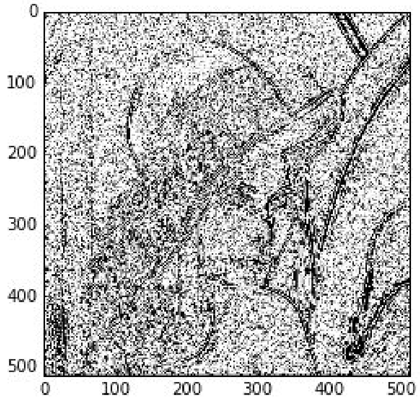


The Laplacian operator

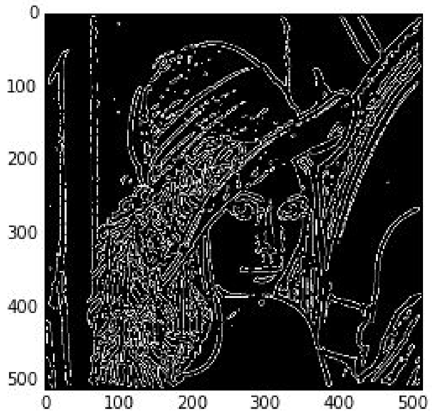


Laplacian without smoothed

The Laplacian operator



Laplacian without smoothed



Marr-H with 15x15 template
sigma = 2.3

LoG operator

LoG operator (Marr-Hildreth operator)

Laplacian of Gaussian operator

$$\nabla^2(g(x,y) * P) \;=\; \underline{\nabla^2(g(x,y))} * P$$

$$\mathbf{g(x, y)}$$

$$g(x) = \frac{1}{\sigma\sqrt{2\pi}} * exp\{-\frac{(x-\mu)^2}{2\sigma^2}\}$$

$$g(x) = a * exp\{-\frac{(x-\mu)^2}{2\sigma^2}\}$$

$$g(x, y) = a * exp\{-\frac{(x-\mu_x)^2 + (y-\mu_y)^2}{2\sigma^2}\}$$

$$g(x, y) = a * exp\{-\frac{x^2 + y^2}{2\sigma^2}\}$$

$\mathbf{\bigtriangledown^2 g(x, y)}$

$$\frac{\partial}{\partial x}g(x,y) = \frac{\partial}{\partial x}a * exp\{-\frac{x^2+y^2}{2\sigma^2}\} = -\frac{x}{\sigma^2}exp\{-\frac{x^2+y^2}{2\sigma^2}\}$$

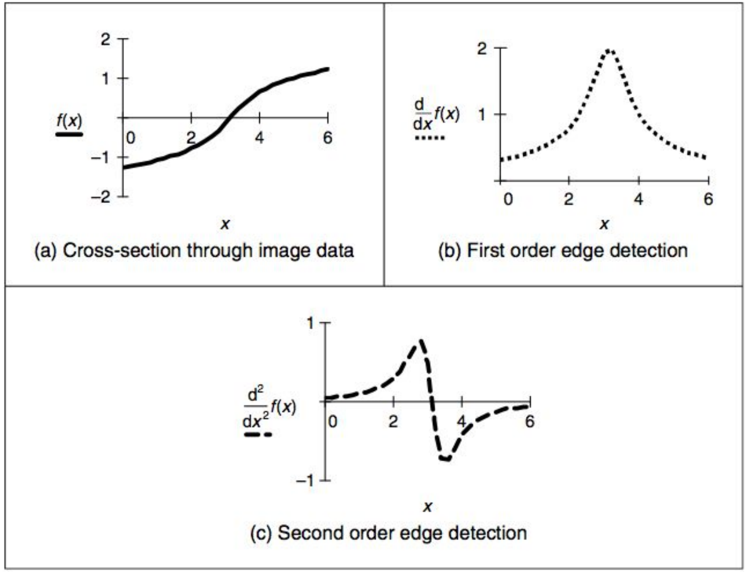
$$\frac{\partial^2}{\partial^2 x}g(x,y) = a * \frac{x^2-\sigma^2}{\sigma^4} * exp\{-\frac{x^2+y^2}{2\sigma^2}\}$$

$$\frac{\partial^2}{\partial^2 y}g(x,y) = a * \frac{y^2-\sigma^2}{\sigma^4} * exp\{-\frac{x^2+y^2}{2\sigma^2}\}$$

$$\nabla^{\mathbf{2}}\mathbf{g}(\mathbf{x}, \mathbf{y})$$

$$\begin{aligned} LoG &= \nabla^2 g(x, y) \\ &= \frac{\partial^2}{\partial^2 x} g(x, y) + \frac{\partial^2}{\partial^2 y} g(x, y) \\ &= \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} * exp\{-\frac{x^2 + y^2}{2\sigma^2}\} \end{aligned}$$

Zero Cross

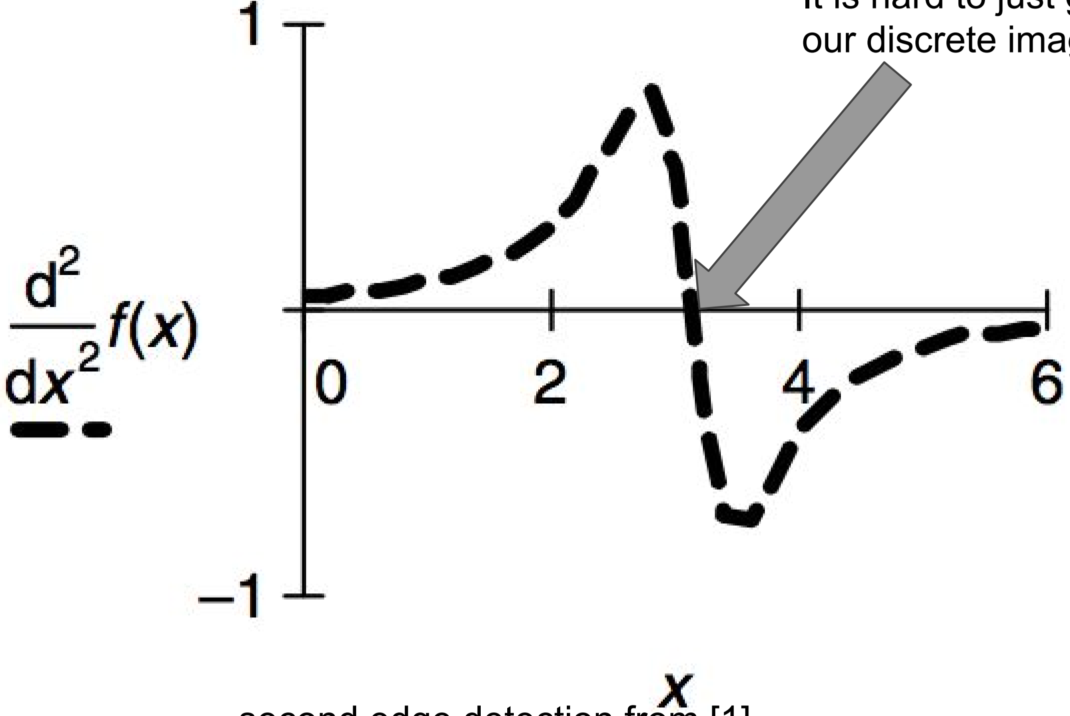


first and second edge detection from [1]

Ref.: Nixon M. Feature extraction & image processing[M]. Academic Press, 2008.

Zero Cross

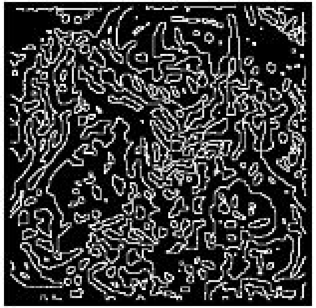
It is hard to just get 0 in
our discrete image



second edge detection from [1]

Ref.: Nixon M. Feature extraction & image processing[M]. Academic Press, 2008.

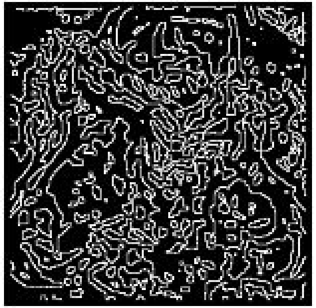
Comparison of LoG



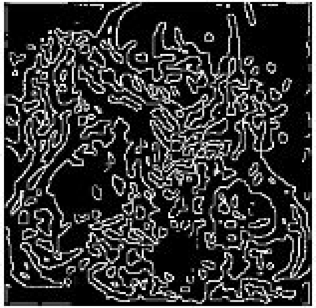
LoG

Method	Time cost (s)
Gaussian+Laplasian	0.7
LoG	0.35

Comparison of LoG



LoG

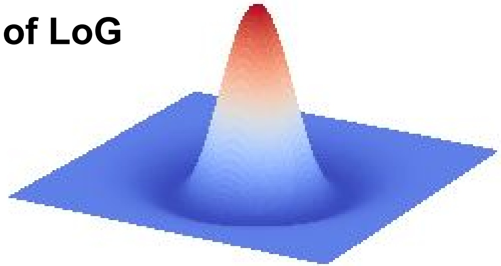


Fourier of LoG

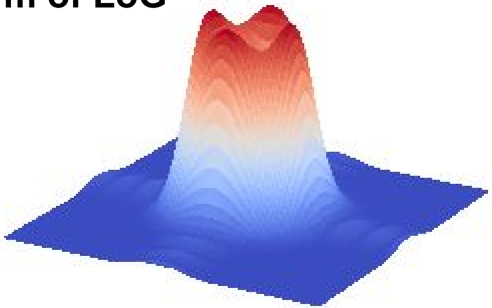
Method	Time cost (s)
Gaussian+Laplasian	0.7
LoG	0.35
Fourier of LoG	0.015

Fourier of LoG

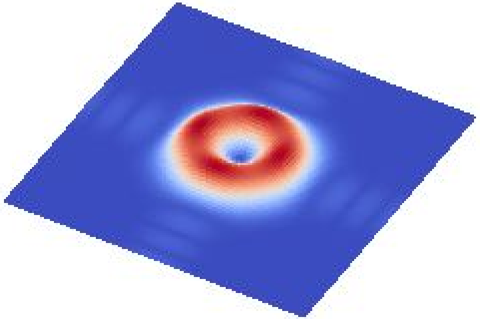
Original Shape of LoG



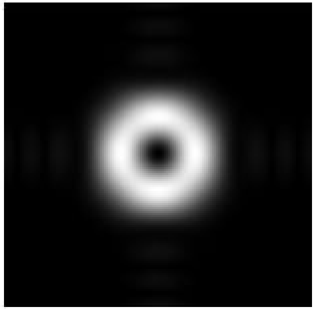
Fourier Transform of LoG



side-looking

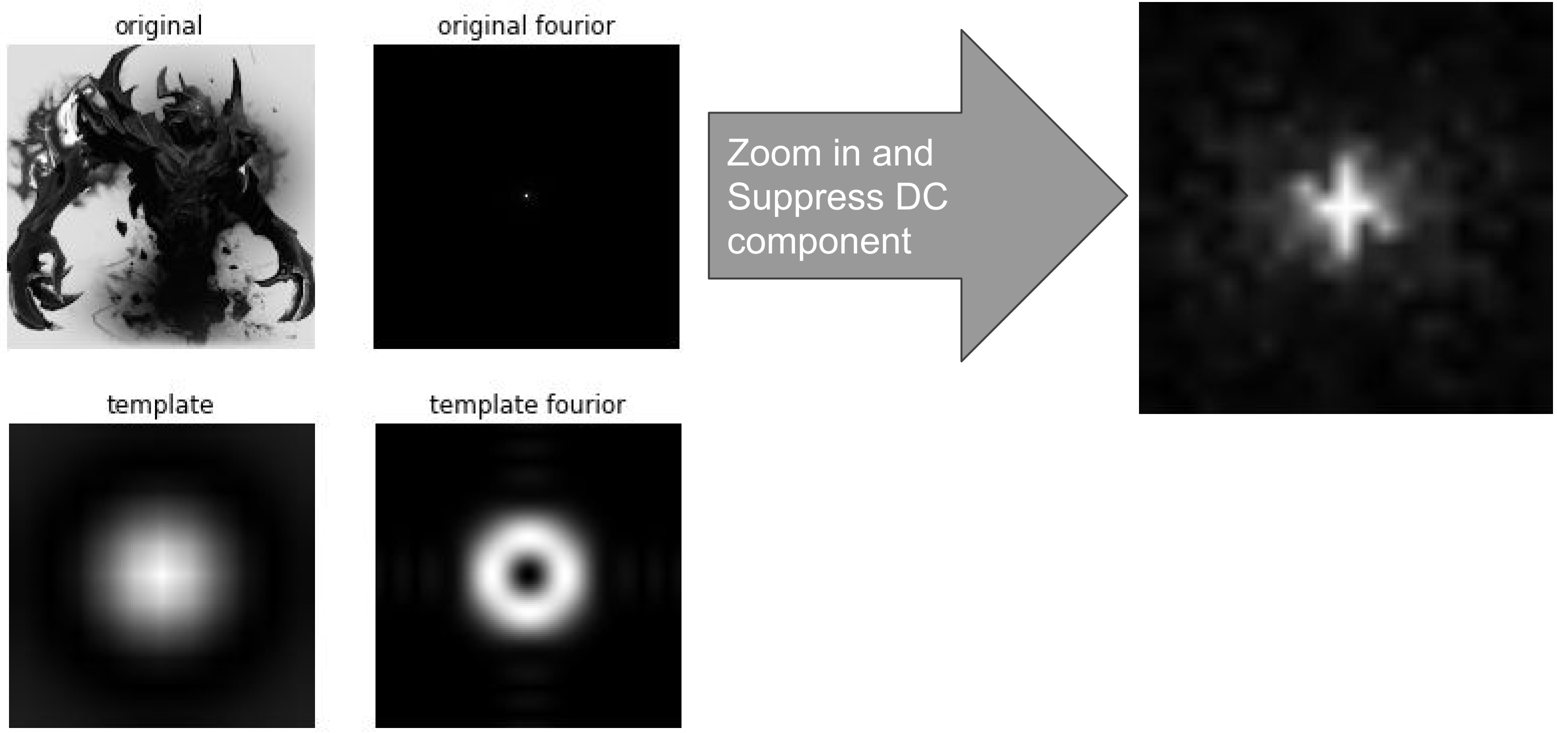


overlook

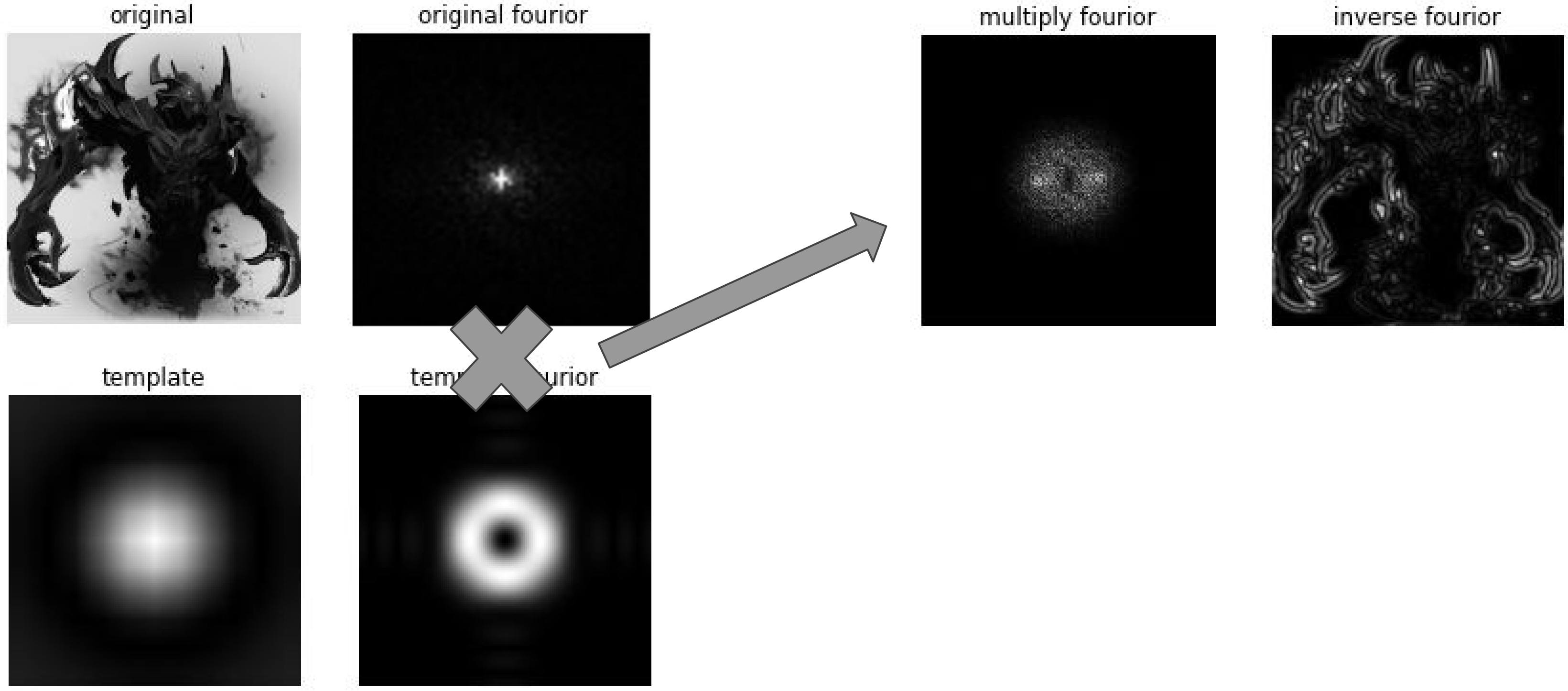


2D shape

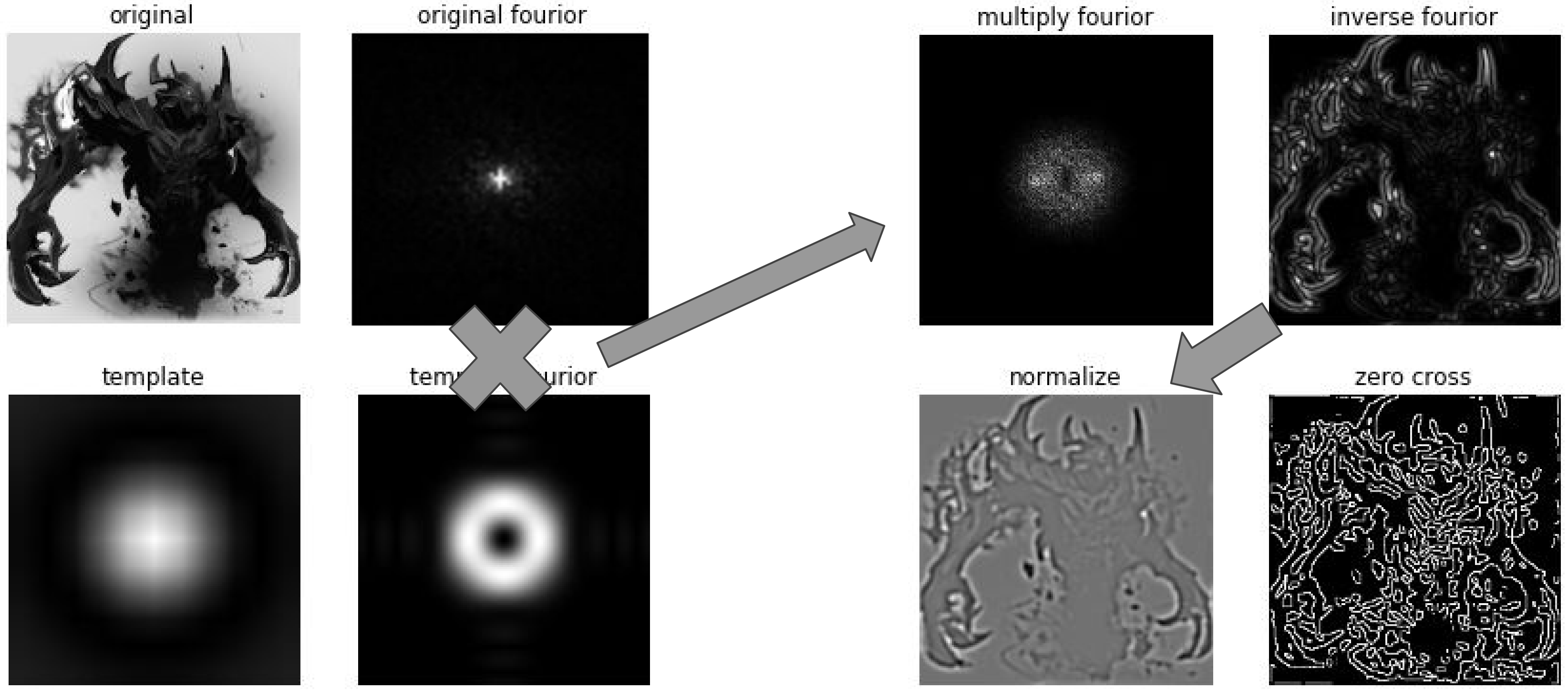
$$Image * Template = F^{-1}\{F(Image)F(Template)\}$$



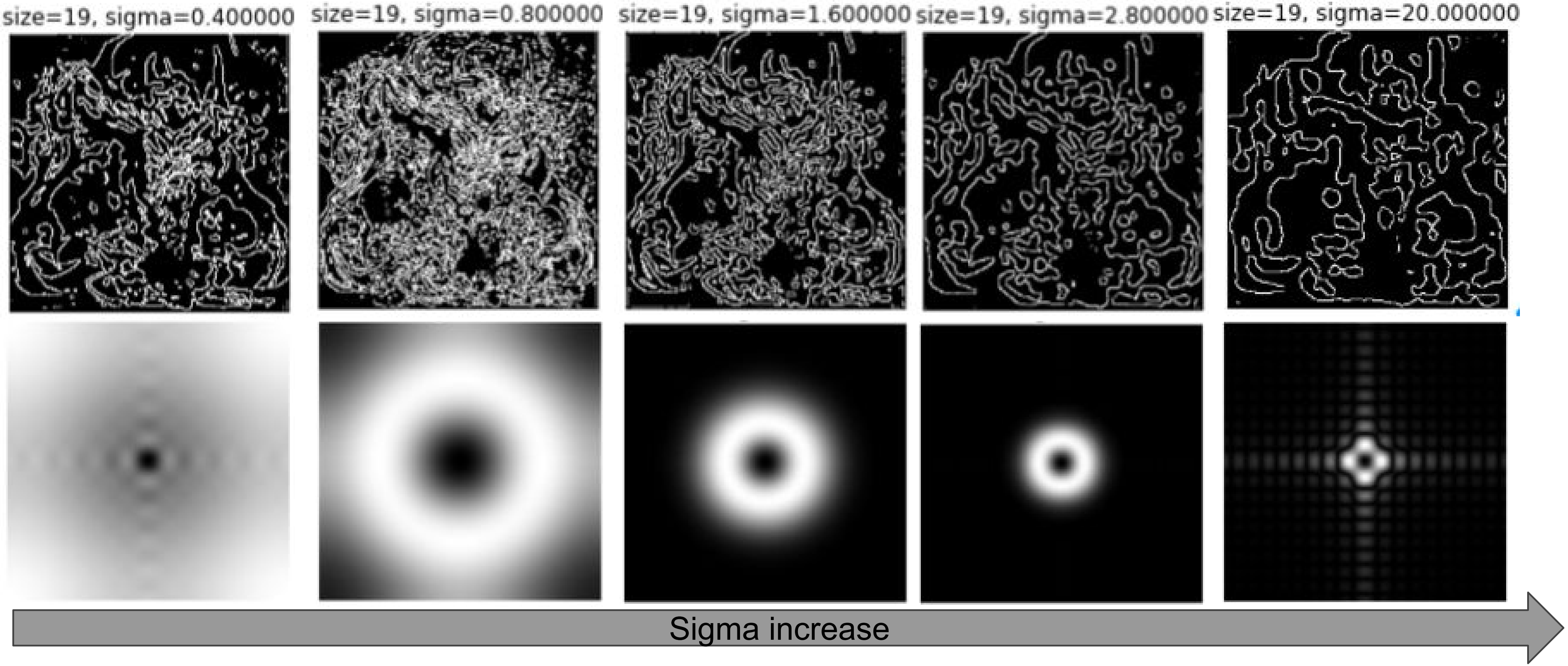
$$Image * Template = F^{-1}\{F(Image)F(Template)\}$$



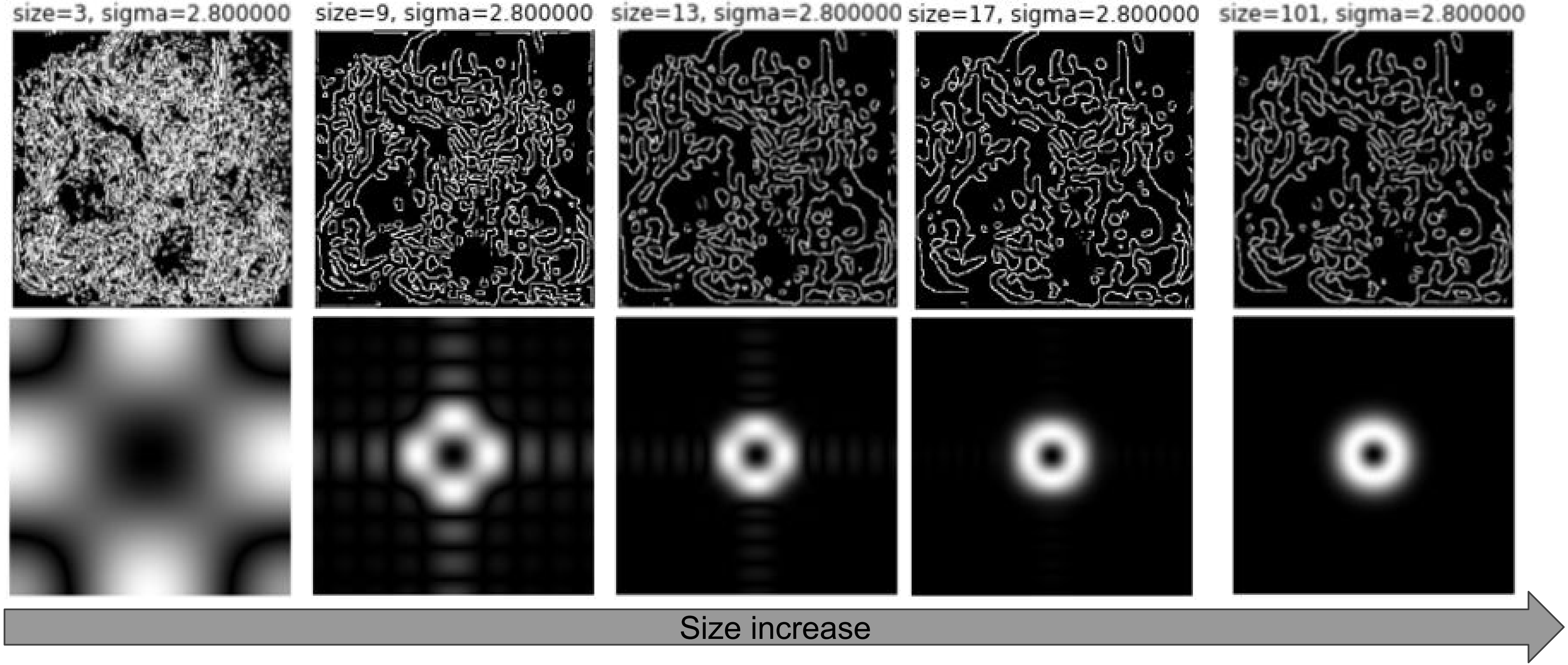
$$Image * Template = F^{-1}\{F(Image)F(Template)\}$$



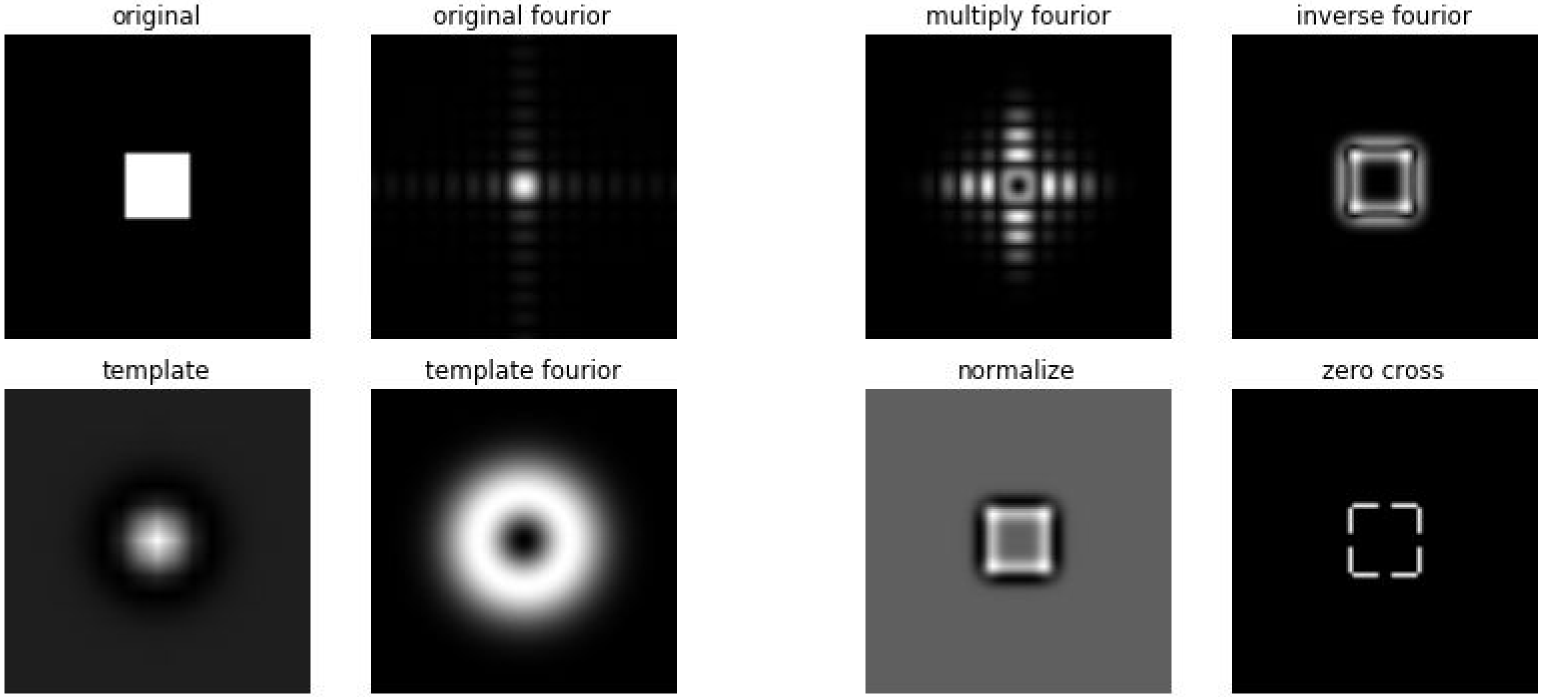
Influence of Sigma



Influence of Size



Simpler example, Fourier is extremely beautiful!



Thanks!
Any Question?

Reference list

[1] Nixon M. Feature extraction & image processing[M]. Academic Press, 2008.

[2] Edge detection, http://blog.csdn.net/xiaowei_cqu/article/details/7829481

[3] Laplacian of Gaussian (LoG), http://fourier.eng.hmc.edu/e161/lectures/gradient/node8.html

[4] Laplacian Operator, http://blog.csdn.net/wangxiaojun911/article/details/7420965

[5] Wang, Xin. "Laplacian operator-based edge detectors." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 29.5 (2007): 886-890.

[6] Sharifi, Mohsen, Mahmoud Fathy, and Maryam Tayefeh Mahmoudi. "A classified and comparative study of edge detection algorithms." *Information Technology: Coding and Computing, 2002. Proceedings. International Conference on*. IEEE, 2002.

[7] Haralick, Robert M. "Digital step edges from zero crossing of second directional derivatives." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 1 (1984): 58-68.