

图像处理与分析

— 图像边缘检测

授课教师：孙剑

jiansun@mail.xjtu.edu.cn

<http://jiansun.gr.xjtu.edu.cn>

西安交通大学 数学与统计学院

目录

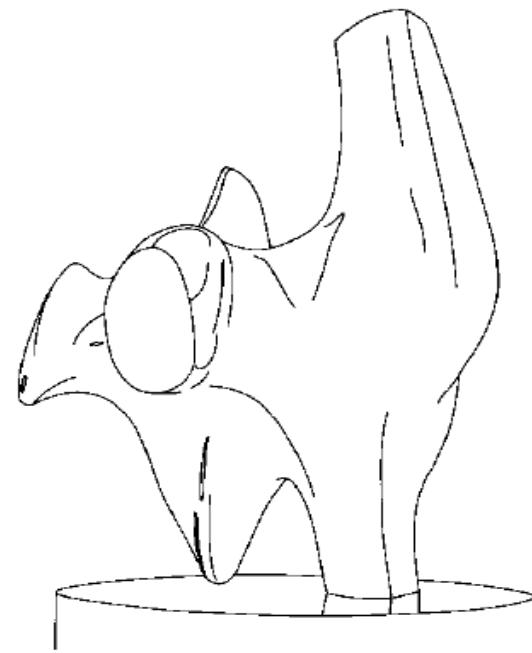
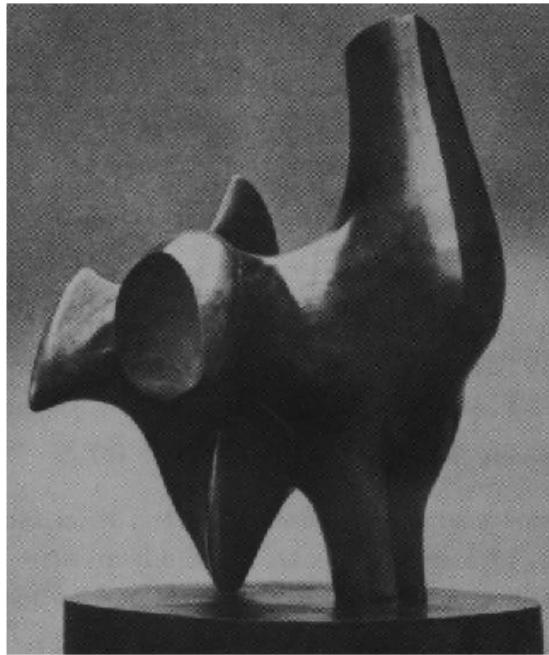
- 图像边缘检测的基本问题
- 图像边缘检测梯度算子
- 图像边缘检测方法
- 图像边缘检测的多尺度方法

图像边缘检测问题

- 图像边缘(**edge**): 图像中亮度变化比较大的点。图像边缘点出现一般是由于深度的不连续、物体表面方向的变换、物体属性变化或者场景照明变化引起。
- 边缘检测: 自动检测出图像的边缘点或线。在图像分析和理解中有重要应用

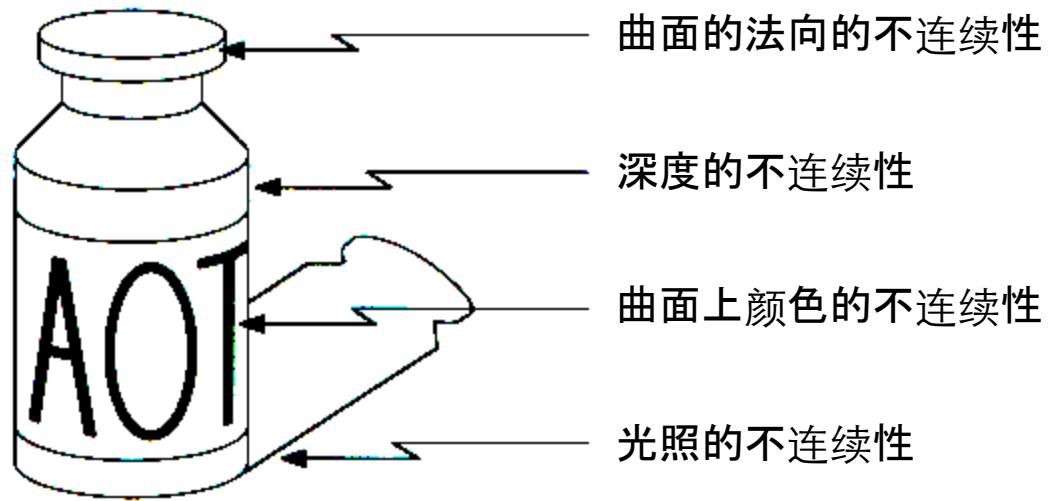


图像边缘检测问题



- 将2维图像转换为一系列曲线的集合
 - 获得场景中的显著特征
 - 获得比像素点更为精炼的表达

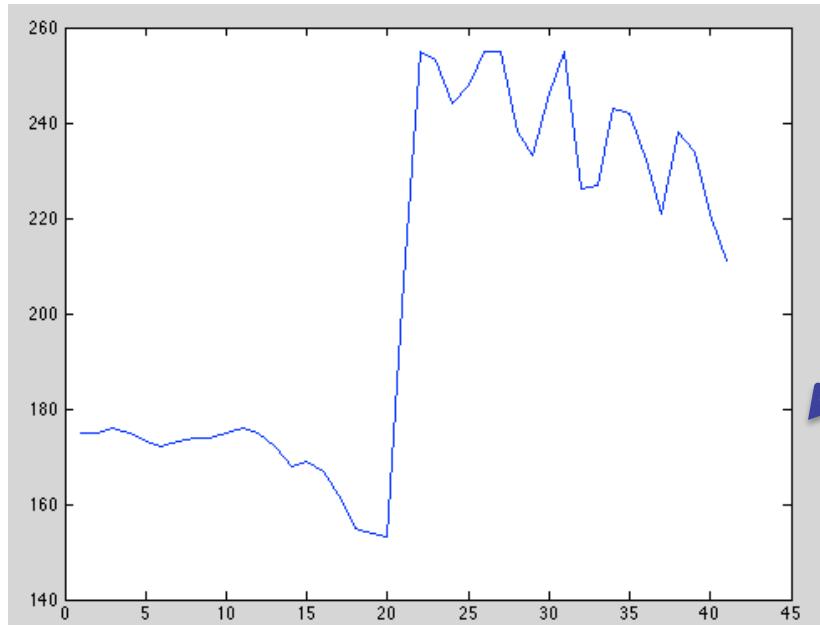
图像边缘的产生机制



- 图像边缘的产生由多种机制引起

图像边缘类型

- 图像边缘(**edge**): 灰度变换剧烈

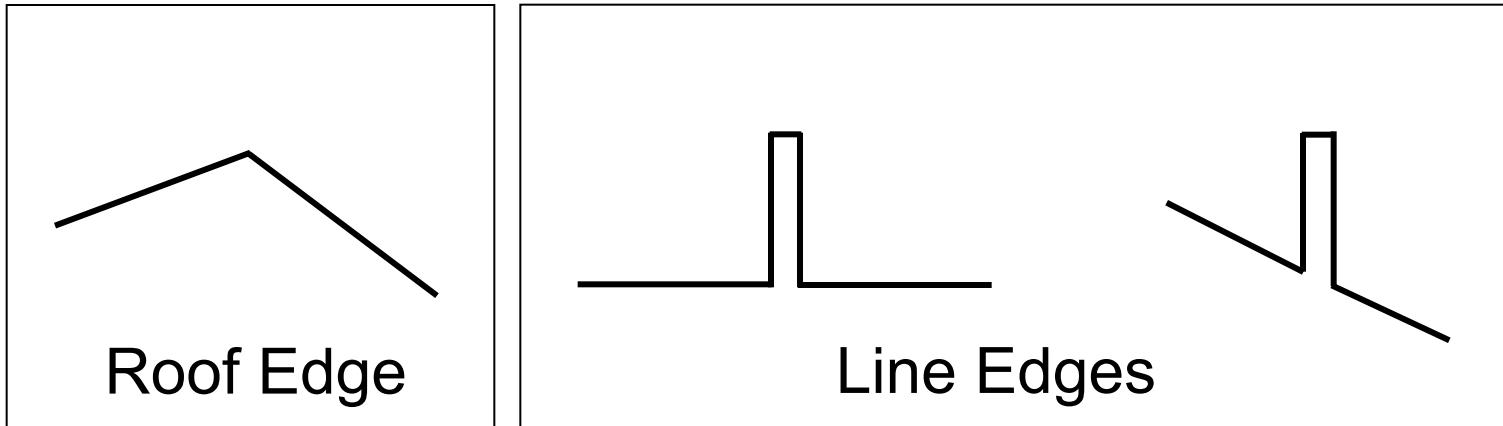
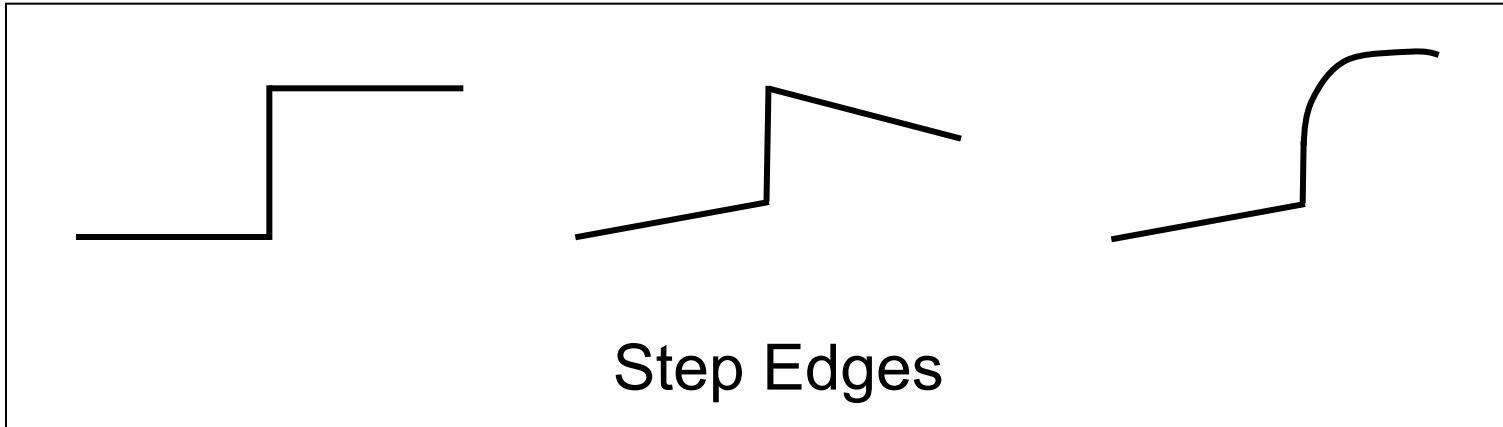


图像边缘模型:

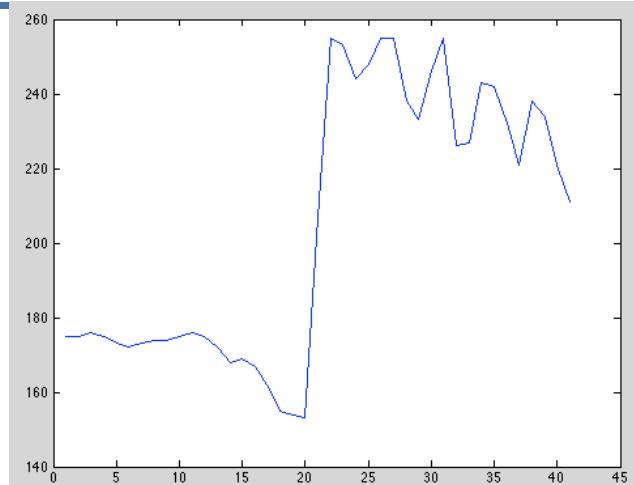
$$f(x) = \frac{I_r - I_i}{2} \left(\text{erf}\left(\frac{x}{\sqrt{2}\sigma}\right) + 1 \right) + I_L$$

图像边缘类型

- 图像边缘的类型



图像边缘类型



真实的图像边缘:含有噪声, 不一定符合理想的边缘模型

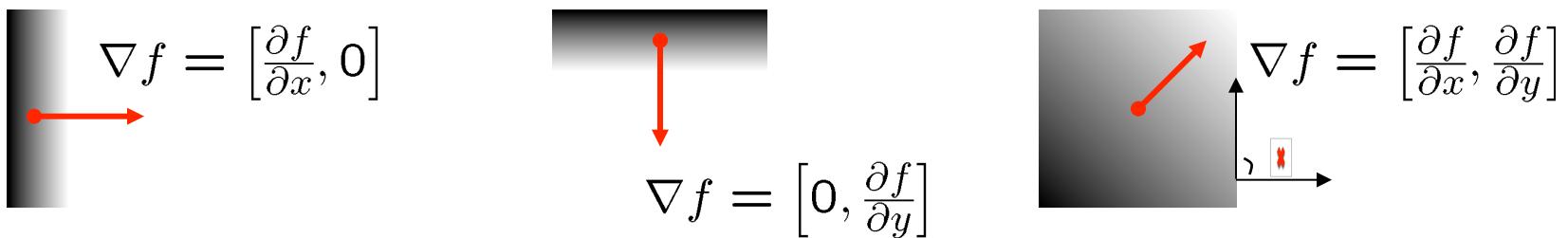
边缘检测算子的设计目标:

- 可检测图像边缘强度
- 可检测图像边缘的方向
- 具有高的检测率和好的边缘定位能力

图像边缘类型

- 图像梯度计算公式: $\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]$

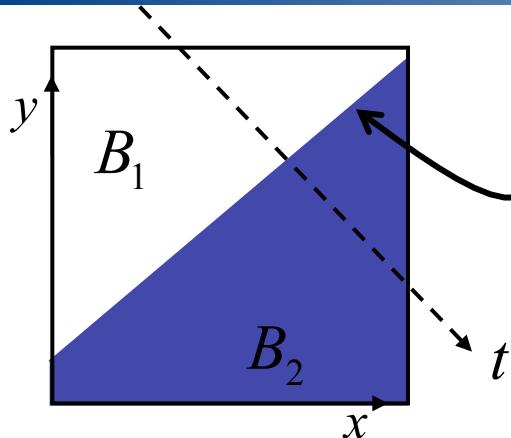
图像梯度反应了图像梯度下降的最快方向



- 梯度方向 $\theta = \tan^{-1} \left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x} \right)$
- 边缘的强度由梯度的模测度:

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

图像边缘理想模型



理想的边缘

$$L(x, y) = x \sin \theta - y \cos \theta + \rho = 0$$

$$B_1 : L(x, y) < 0$$

$$B_2 : L(x, y) > 0$$

单位阶段函数(Unit step function):

$$u(t) = \begin{cases} 1 & \text{for } t > 0 \\ \frac{1}{2} & \text{for } t = 0 \\ 0 & \text{for } t < 0 \end{cases} \quad u(t) = \int_{-\infty}^t \delta(s) ds$$

理想边缘图像的灰度模型:

$$I(x, y) = B_1 + (B_2 - B_1)u(x \sin \theta - y \cos \theta + \rho)$$

图像边缘理想模型

- 理想边缘图像的灰度模型:

$$I(x, y) = B_1 + (B_2 - B_1) \mu(x \sin \theta - y \cos \theta + \rho)$$

- 梯度:

$$\frac{\partial I}{\partial x} = +\sin \theta (B_2 - B_1) \delta(x \sin \theta - y \cos \theta + \rho)$$

$$\frac{\partial I}{\partial y} = -\cos \theta (B_2 - B_1) \delta(x \sin \theta - y \cos \theta + \rho)$$

- 梯度强度:

$$s(x, y) = \left(\frac{\partial I}{\partial x} \right)^2 + \left(\frac{\partial I}{\partial y} \right)^2 = [(B_2 - B_1) \delta(x \sin \theta - y \cos \theta + \rho)]^2$$

边缘梯度模: $\sqrt{s(x, y)}$

边缘的方向: $\arctan \left(\frac{\partial I}{\partial y} / \frac{\partial I}{\partial x} \right)$ (边缘的法向方向)

图像边缘理想模型

- 理想边缘图像的灰度模型:

$$I(x, y) = B_1 + (B_2 - B_1) \mu(x \sin \theta - y \cos \theta + \rho)$$

- 梯度:

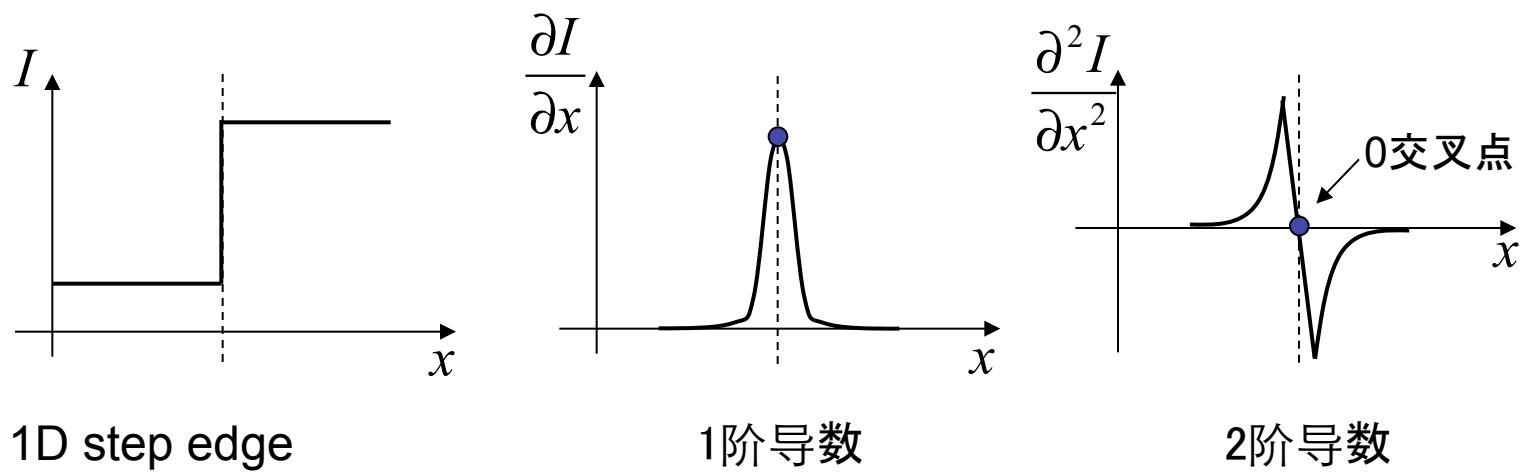
$$\frac{\partial I}{\partial x} = +\sin \theta (B_2 - B_1) \delta(x \sin \theta - y \cos \theta + \rho)$$

$$\frac{\partial I}{\partial y} = -\cos \theta (B_2 - B_1) \delta(x \sin \theta - y \cos \theta + \rho)$$

- Laplacian:

$$\nabla^2 I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} = (B_2 - B_1) \delta'(x \sin \theta - y \cos \theta + \rho)$$

图像边缘理想模型



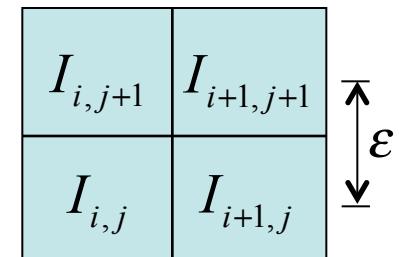
图像微分算子

● 图像微分的计算

差分算法进行逼近:

$$\frac{\partial I}{\partial x} \approx \frac{1}{2\varepsilon} ((I_{i+1,j+1} - I_{i,j+1}) + (I_{i+1,j} - I_{i,j}))$$

$$\frac{\partial I}{\partial y} \approx \frac{1}{2\varepsilon} ((I_{i+1,j+1} - I_{i+1,j}) + (I_{i,j+1} - I_{i,j}))$$



卷积核:

$$\frac{\partial I}{\partial x} \approx \frac{1}{2\varepsilon} \begin{array}{|c|c|} \hline -1 & 1 \\ \hline -1 & 1 \\ \hline \end{array}$$

$$\frac{\partial I}{\partial y} \approx \frac{1}{2\varepsilon} \begin{array}{|c|c|} \hline 1 & 1 \\ \hline -1 & -1 \\ \hline \end{array}$$

图像微分算子

- 二阶偏导数：

$$\frac{\partial^2 I}{\partial x^2} \approx \frac{1}{\varepsilon^2} (I_{i-1,j} - 2I_{i,j} + I_{i+1,j})$$

$$\frac{\partial^2 I}{\partial y^2} \approx \frac{1}{\varepsilon^2} (I_{i,j-1} - 2I_{i,j} + I_{i,j+1})$$

$I_{i-1,j+1}$	$I_{i,j+1}$	$I_{i+1,j+1}$
$I_{i-1,j}$	$I_{i,j}$	$I_{i+1,j}$
$I_{i-1,j-1}$	$I_{i,j-1}$	$I_{i+1,j-1}$

- 拉普拉斯算子：

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

卷积核：

$$\nabla^2 I \approx \frac{1}{\varepsilon^2} \begin{array}{|c|c|c|} \hline 0 & 1 & 0 \\ \hline 1 & -4 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array}$$

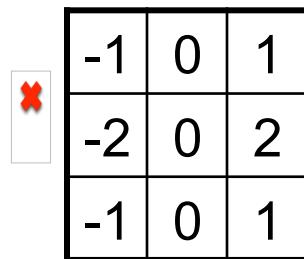
或

$$\frac{1}{6\varepsilon^2} \begin{array}{|c|c|c|} \hline 1 & 4 & 1 \\ \hline 4 & -20 & 4 \\ \hline 1 & 4 & 1 \\ \hline \end{array}$$

图像微分算子

- Sobel算子：

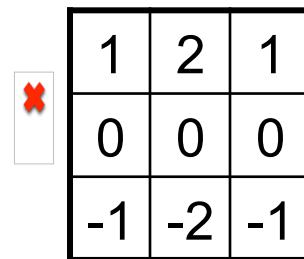
卷积核：



A 3x3 kernel for horizontal edge detection. It has values -1, 0, 1 in the top row, -2, 0, 2 in the middle row, and -1, 0, 1 in the bottom row. A red 'X' is placed in the first column of the first row.

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

$$\frac{\partial I}{\partial x}$$



A 3x3 kernel for vertical edge detection. It has values 1, 2, 1 in the top row, 0, 0, 0 in the middle row, and -1, -2, -1 in the bottom row. A red 'X' is placed in the first column of the first row.

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

$$\frac{\partial I}{\partial y}$$

图像微分算子

- Robert算子



1	0
0	-1

0	1
-1	0



无法

图像微分算子

- Prewitt算子：

卷积核：

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

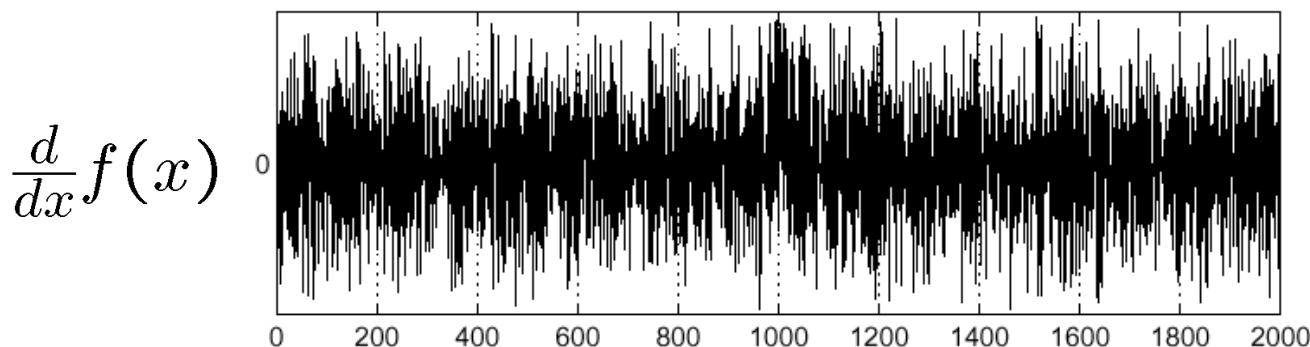
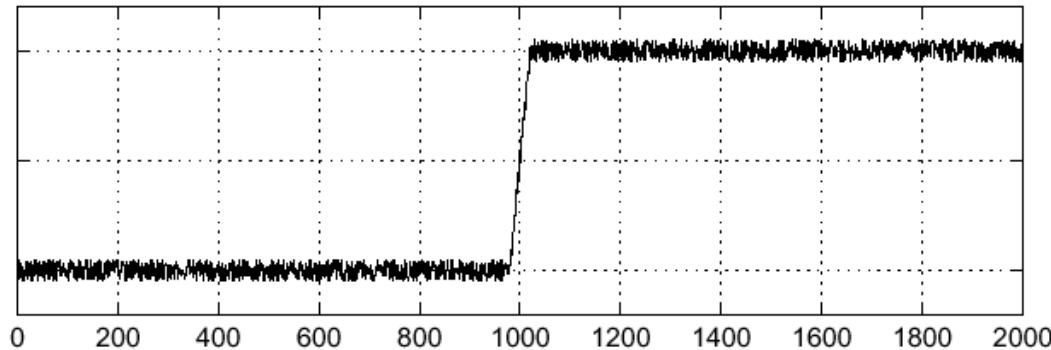
$$\frac{\partial I}{\partial x}$$

1	1	1
0	0	0
-1	-1	1

$$\frac{\partial I}{\partial y}$$

图像边缘检测

图像边缘检测：先光滑化 → 再求微分

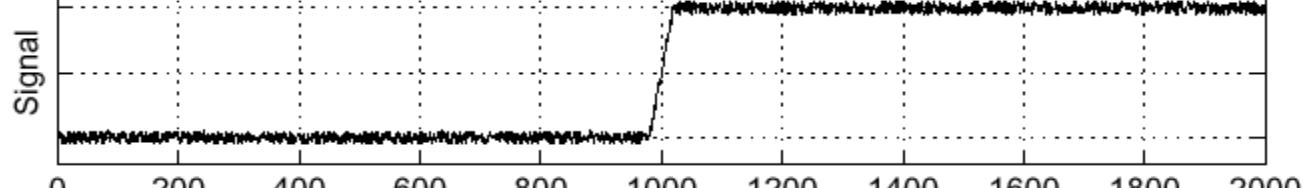


图像微分计算受到噪声干扰

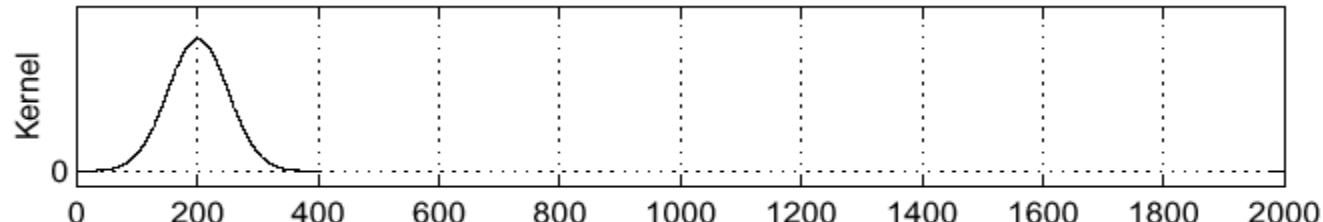
图像边缘检测



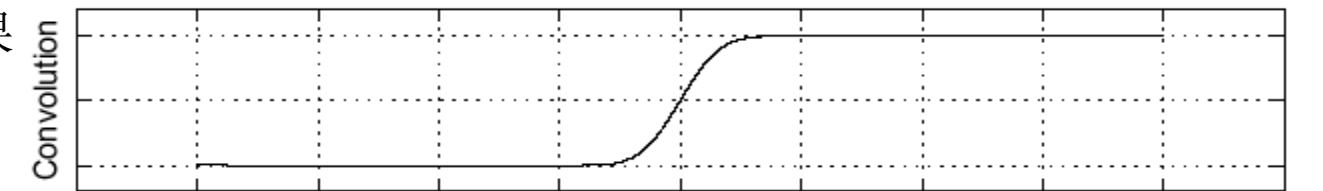
Sigma = 50



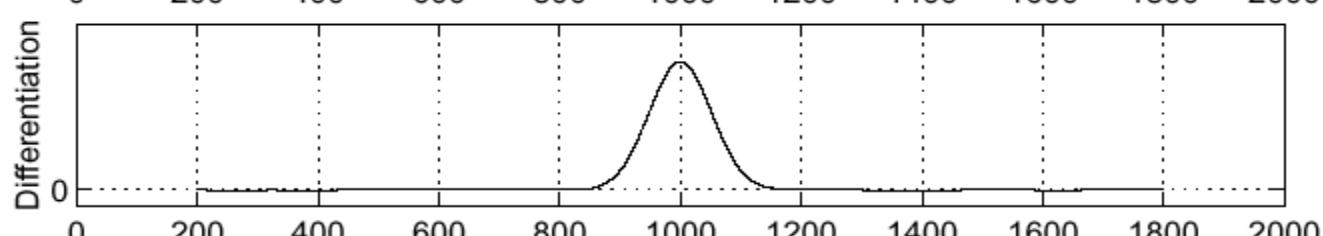
信号高斯滤波



高斯滤波结果



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



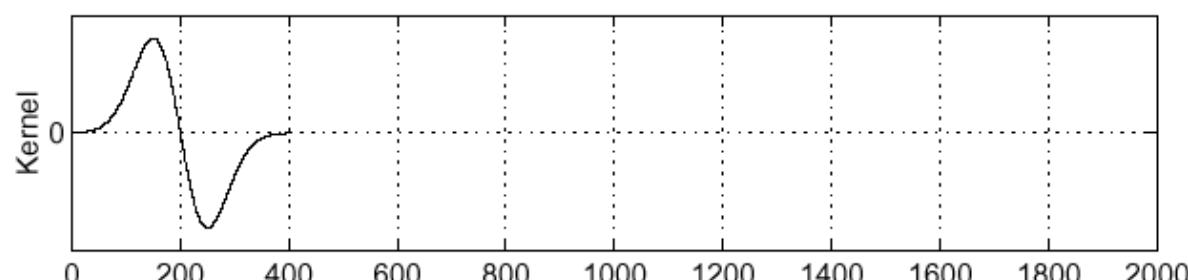
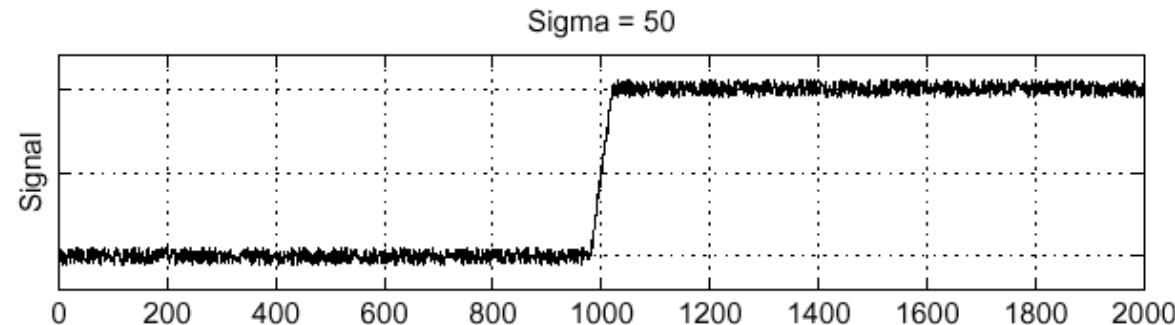
边缘点位于梯度的极值点 : $\frac{\partial}{\partial x}(h \star f)$

图像边缘检测

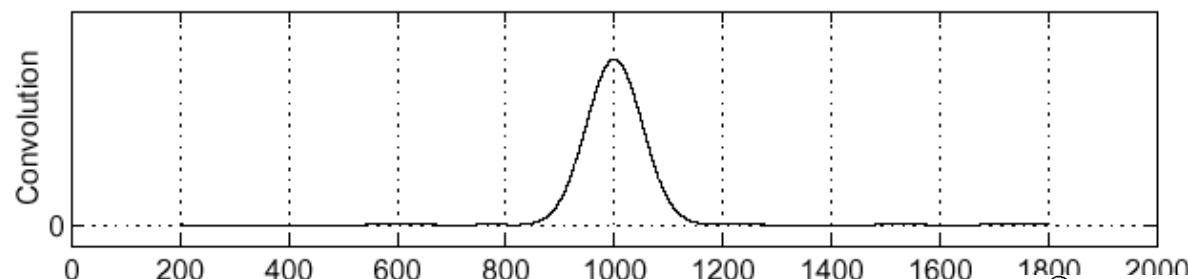
$$\frac{\partial}{\partial x}(h \star f) = (\frac{\partial}{\partial x}h) \star f$$



信号滤波：滤波器
为高斯导数函数



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



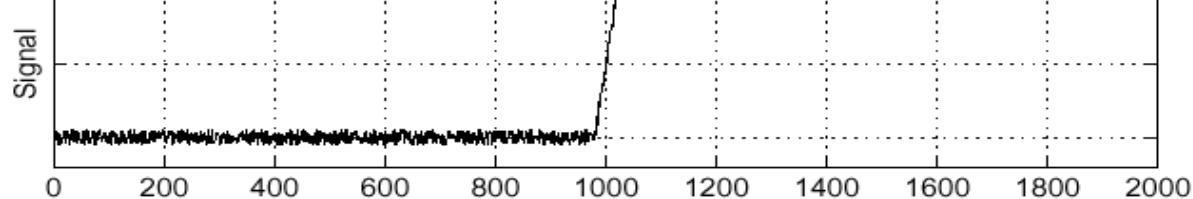
边缘点位于梯度的极值点： $\frac{\partial}{\partial x}(h \star f)$

图像边缘检测

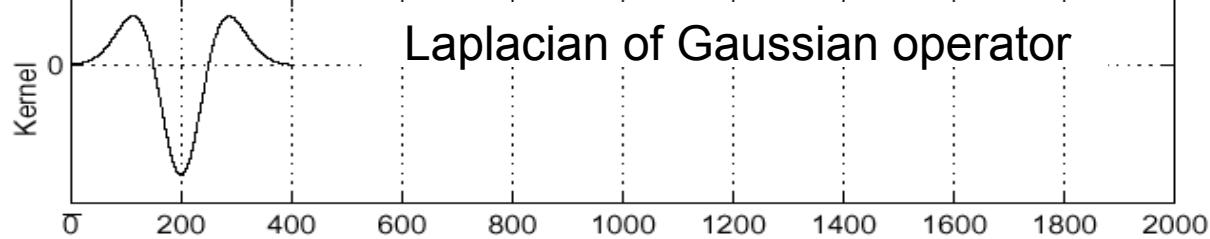
$$\frac{\partial^2}{\partial x^2}(h * f) = \left(\frac{\partial^2}{\partial x^2} h \right) * f$$

Laplacian of Gaussian (LoG)

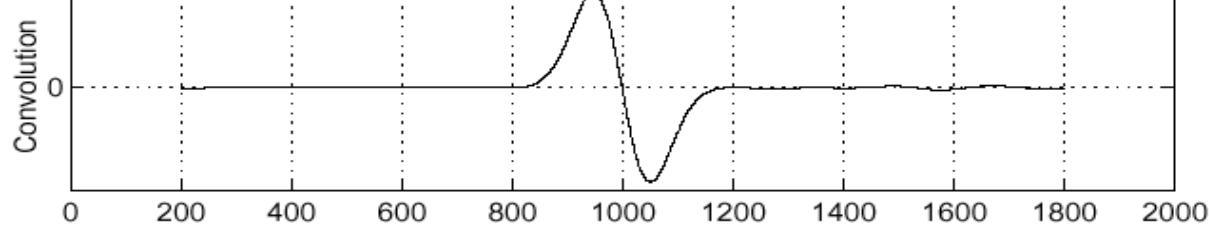
Sigma = 50



$$\frac{\partial^2}{\partial x^2} h$$

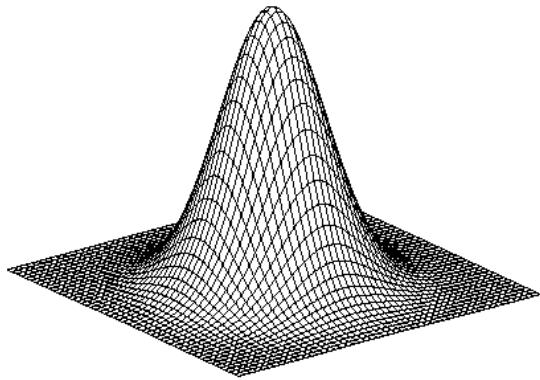


$$\left(\frac{\partial^2}{\partial x^2} h \right) * f$$



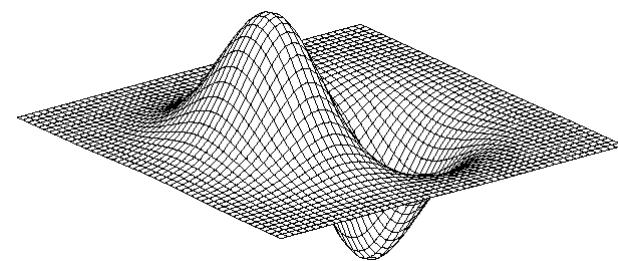
边缘点位于LoG滤波信号的0交叉点

图像边缘检测



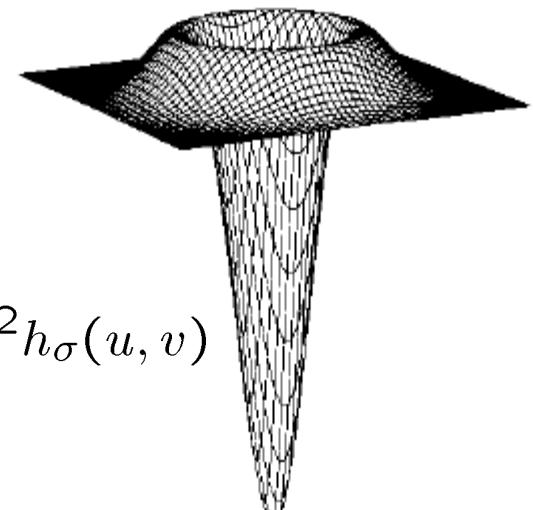
$$h_\sigma(u, v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$

高斯函数



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

高斯导数
Derivative of Gaussian (DoG)



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

高斯的拉普拉斯函数
Laplacian of Gaussian
(LoG)

拉普拉斯:

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

图像边缘检测

- 图像边缘点：Laplacian of Gaussian(LoG)滤波的零交叉点

$$\nabla^2 G \approx G(x, y; \sigma_1) - G(x, y; \sigma_2)$$

$$\nabla^2 G * I$$



$$\sigma = 1$$



$$\sigma = 2$$



$$(b)-(a)$$

图像边缘点：Laplacian of Gaussian(LoG)的零交叉点(zero-crossing)

图像边缘检测

- 图像边缘点：高斯导数(Derivative of Gaussian)滤波的零交叉点

$$\sqrt{\left(\frac{\partial G}{\partial x} * I\right)^2 + \left(\frac{\partial G}{\partial y} * I\right)^2}$$



图像边缘点：图像梯度的局部极大值点

图像边缘检测

基于梯度模阈化的图像边缘检测算法：

(1) Smooth the input image ($\hat{f}(x, y) = f(x, y) * G(x, y)$)

(2) $\hat{f}_x = \hat{f}(x, y) * M_x(x, y)$

(3) $\hat{f}_y = \hat{f}(x, y) * M_y(x, y)$

(4) $magn(x, y) = |\hat{f}_x| + |\hat{f}_y|$

(5) $dir(x, y) = \tan^{-1}(\hat{f}_y / \hat{f}_x)$

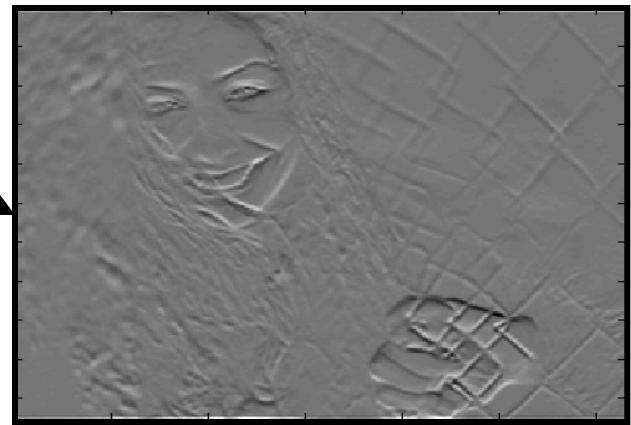
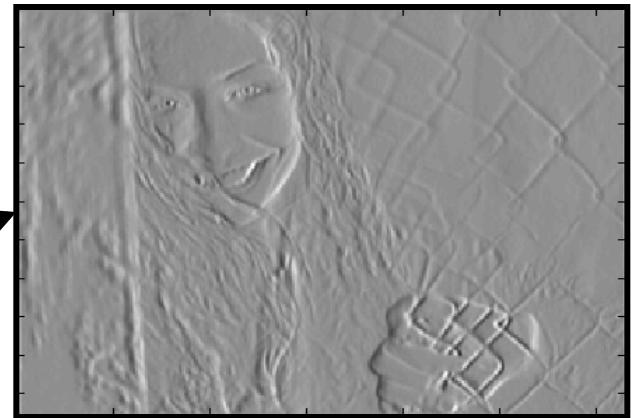
(6) If $magn(x, y) > T$, then possible edge point

图像边缘检测



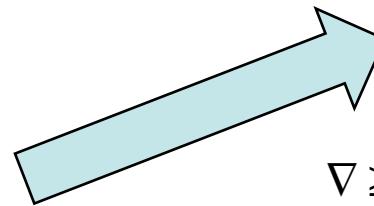
$$\frac{d}{dx} I$$

$$\frac{d}{dy} I$$

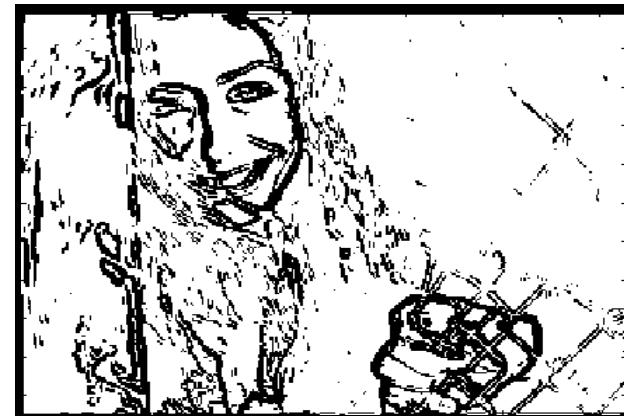
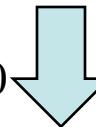


图像边缘检测

$$\nabla = \sqrt{\left(\frac{d}{dx} I \right)^2 + \left(\frac{d}{dy} I \right)^2}$$



$\nabla \geq Threshold = 100$



图像边缘检测

Canny边缘检测算法：

- 对图像进行二维高斯滤波： $G_\sigma * I$
- 对每个像素点，计算局部的边缘法线方向 $\bar{\mathbf{n}} = \frac{\nabla(G * I)}{|\nabla(G * I)|}$
- 计算边缘强度： $|\nabla(G * I)|$
- 非极大消除：获得沿着边缘法线方向的0-交叉点，即边缘强度最大点。
。(该过程称为非极大消除，**non-maximum suppression**)

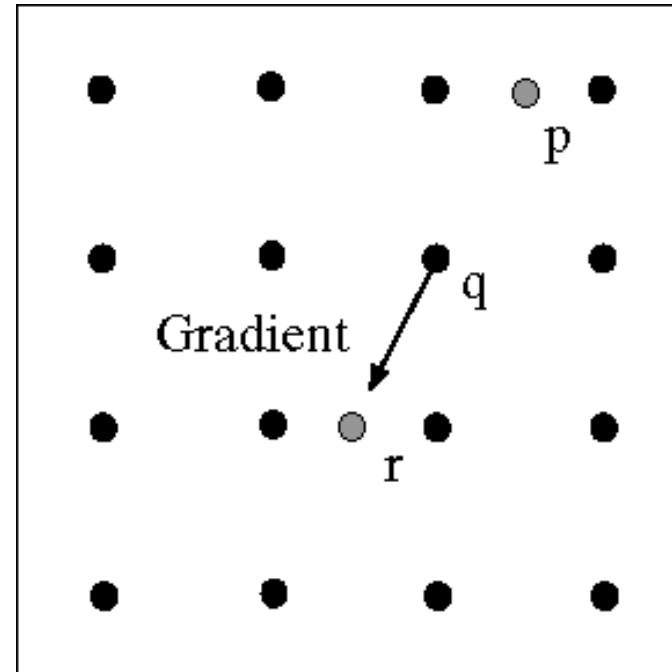
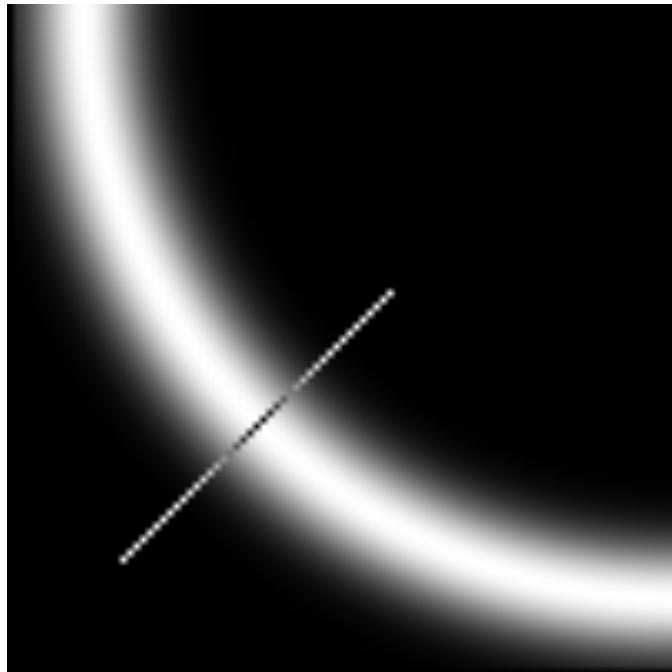
$$\frac{\partial^2(G * I)}{\partial \bar{\mathbf{n}}^2} = 0$$

- 边缘点假设检验

J. Canny, **A Computational Approach To Edge Detection**, IEEE Trans. Pattern Analysis and Machine Intelligence, 8:679-714, 1986.

图像边缘检测

非极大消除， non-maximum suppression



- 检测像素点是否沿着边缘法线方向是局部极大值点
 - 像素p,r点的梯度模值采用插值方法计算

图像边缘检测

边缘点假设检验

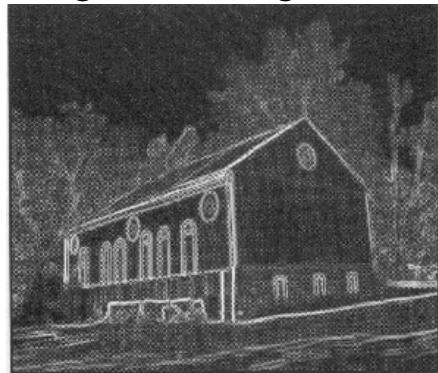
- 标准的阈值化处理:

$$E(x, y) = \begin{cases} 1 & \text{if } \|\nabla f(x, y)\| > T \text{ for some threshold } T \\ 0 & \text{otherwise} \end{cases}$$

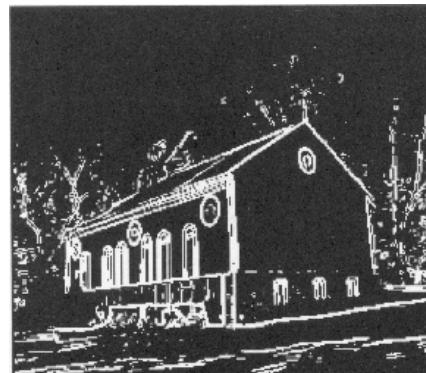
- 可以检测到强边缘点.
- 无法保证边缘点的连续性” .



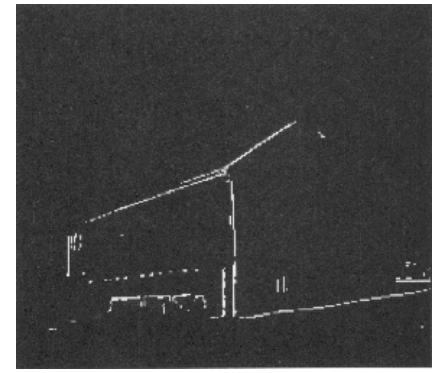
gradient magnitude



low threshold



high threshold



图像边缘检测

- 改进的阈值化处理:

- 低阈值 t_l
- 高阈值 t_h (usually, $t_h = 2t_l$)

$\ \nabla f(x, y)\ \geq t_h$	definitely an edge
$t_l \leq \ \nabla f(x, y)\ < t_h$	maybe an edge, depends on context
$\ \nabla f(x, y)\ < t_l$	definitely not an edge

- 对于第二种情况：“maybe edges”，判断其相邻像素是否是强边缘点。如果是，则标定其为边缘点。

图像Canny边缘检测例子



原始图像

图像Canny边缘检测例子



梯度的模

图像Canny边缘检测例子



非极大消除后结果

图像边缘检测结果对尺度的依赖性



- σ 大可以检测 ~~大~~ 尺度边缘
- σ 小, ~~可~~ 检测出小尺度图像边缘

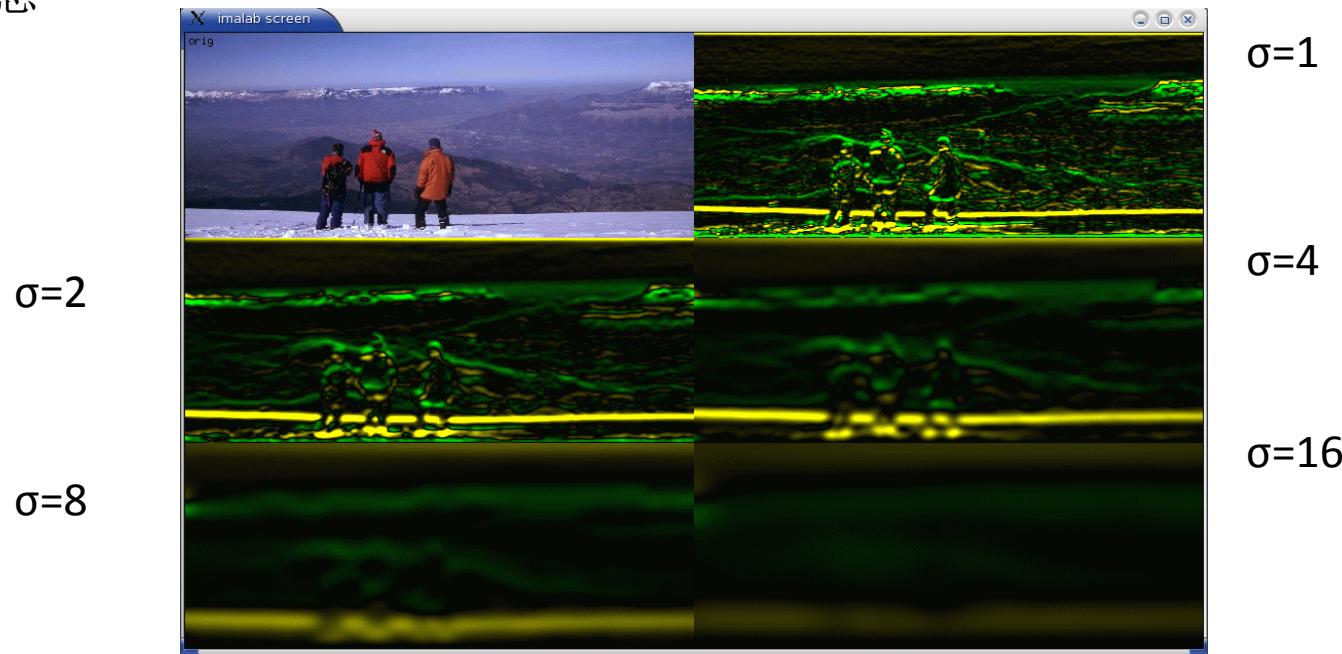


多尺度图像边缘检测

- 尺度空间理论：图像多尺度处理理论，用来在多尺度下描述或处理图像的结构（例如图像边缘）信息
 - 多尺度下图像处理
 - 通过图像结构出现的尺度范围确定显著性结构
 - 在越大尺度范围内均出现的图像结构，认为是显著性结构

多尺度图像边缘检测

将图像用不同尺度(即方差)的高斯滤波器进行滤波，并计算梯度，获得不同尺度下的边缘信息

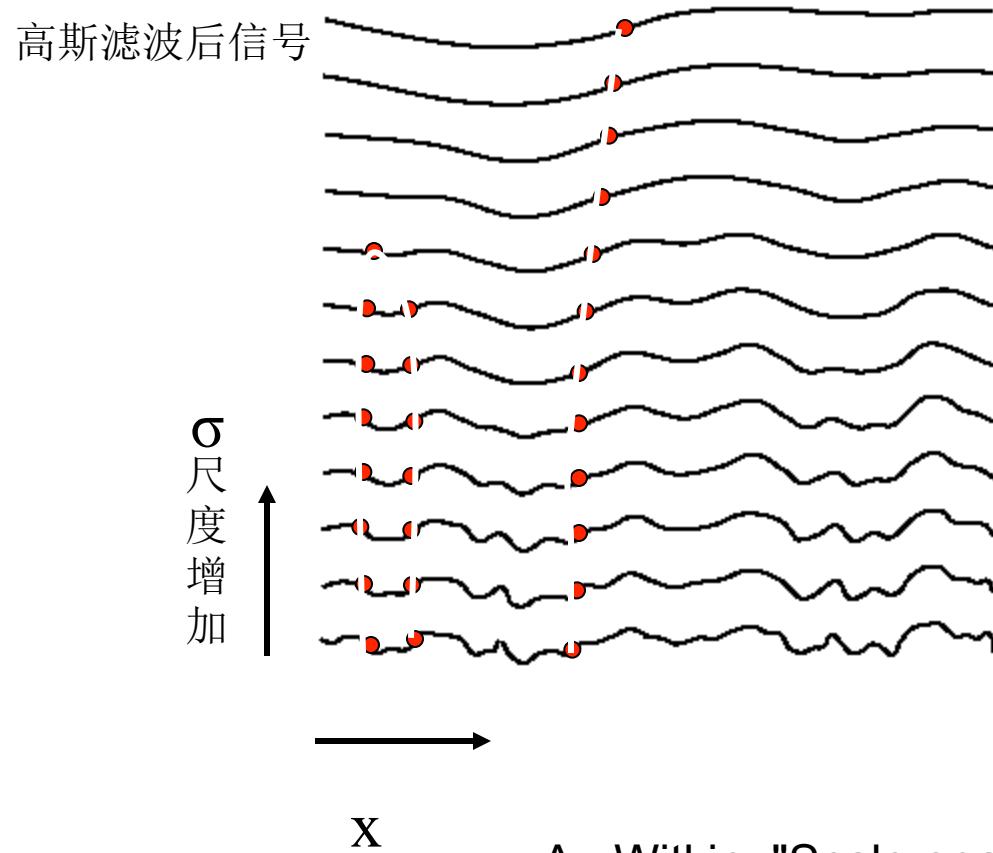


- 关键尺度：图像中重要结构出现的尺度。

例如，在上图中，人在尺度[1.0 - 4.0]范围内均出现，该尺度范围是人的关键尺度范围。

多尺度图像边缘检测

对一维信号进行不同尺度的高斯滤波，在每个尺度下检测零-交叉点（即信号的拐点）

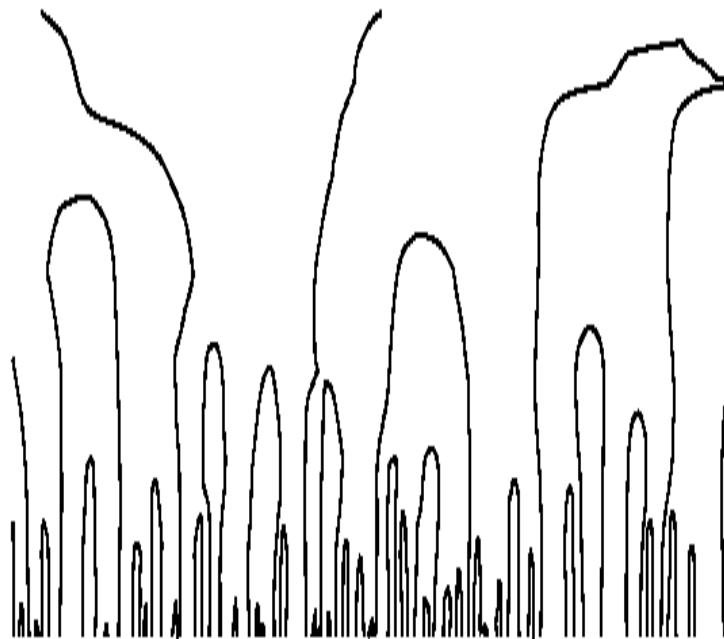


- 检测并画出1D信号的0-交叉点（用红点表示），0-交叉点在一定尺度范围内均存在。
- 在多尺度下跟踪0—交叉点，形成一条轨迹。

A. Witkin, "Scale-space filtering", 8th Int. Joint Conf. Art. Intell., Karlsruhe, Germany, 1019–1022, 1983

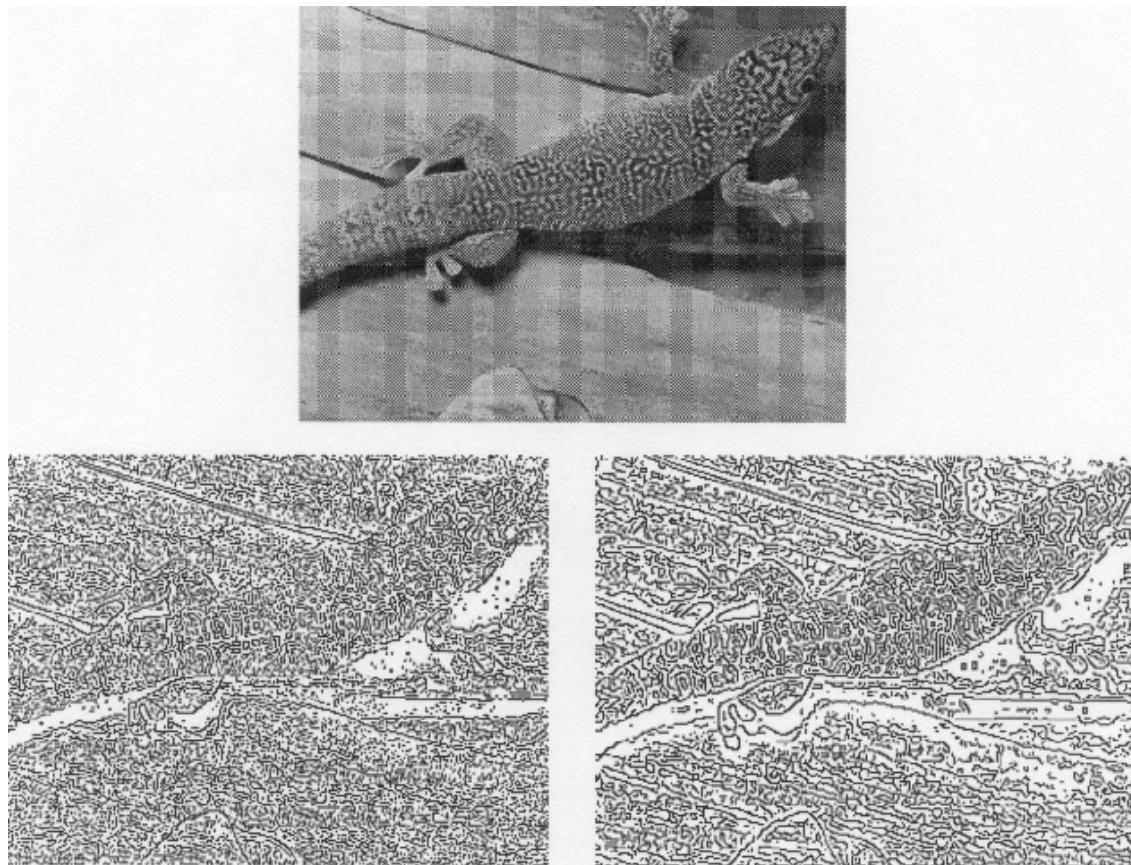
多尺度图像边缘检测

尺度空间性质：



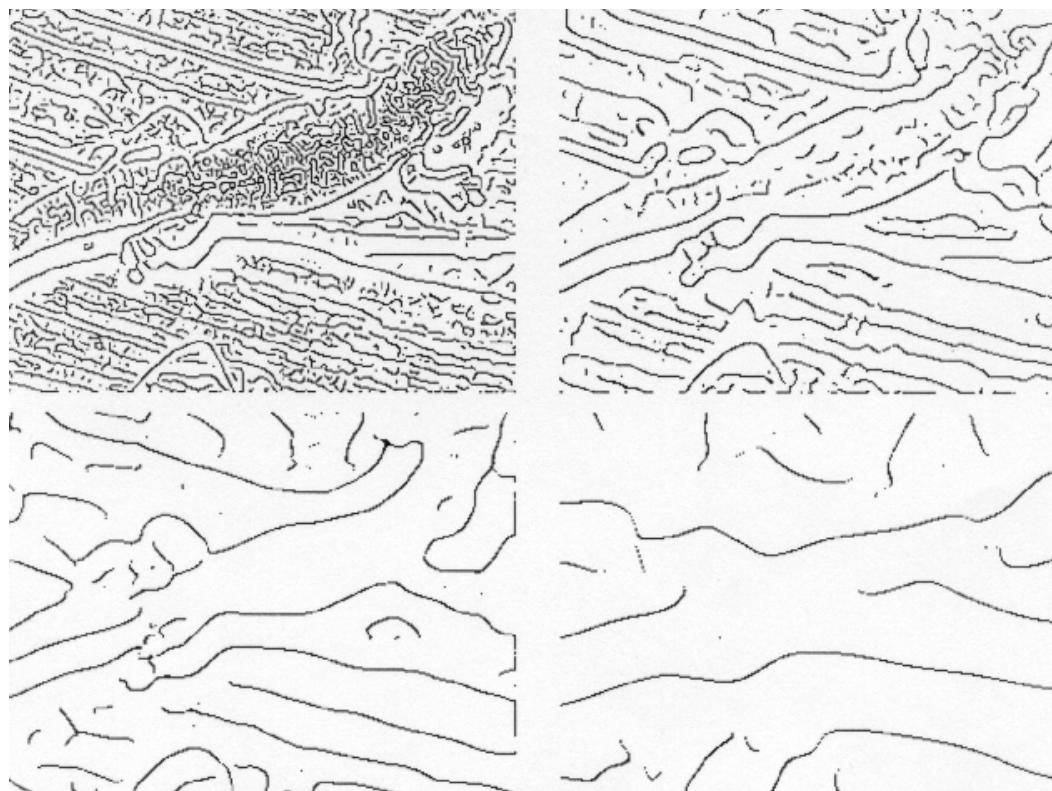
- 高斯滤波形成尺度空间的性质：
 - 0-交叉点随着尺度（即高斯方差）的增加而减少
 - 两个0-交叉点随着尺度增加会趋于合并为一个点.
 - 一条轨迹不会分解为两个或更多的轨迹（即图像随着尺度增加不会出现新的0-交叉点）.

多尺度图像边缘检测



(Canny edges at multiple scales of smoothing, $\sigma=0.5, 1,$

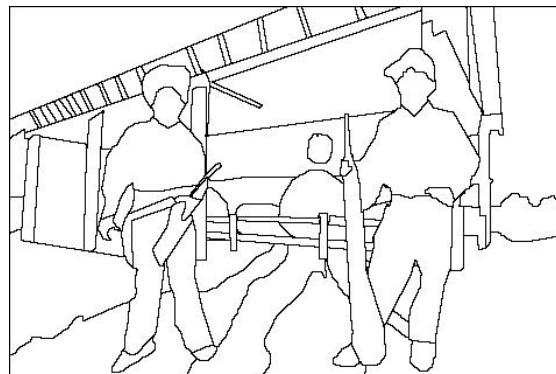
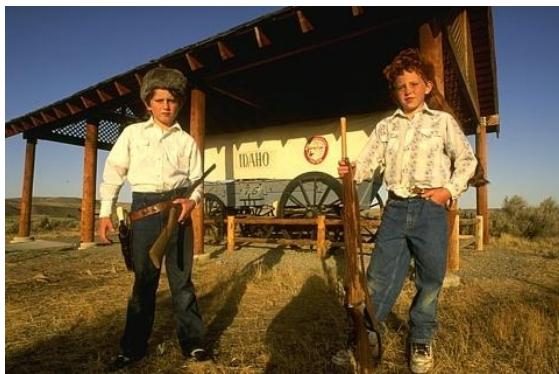
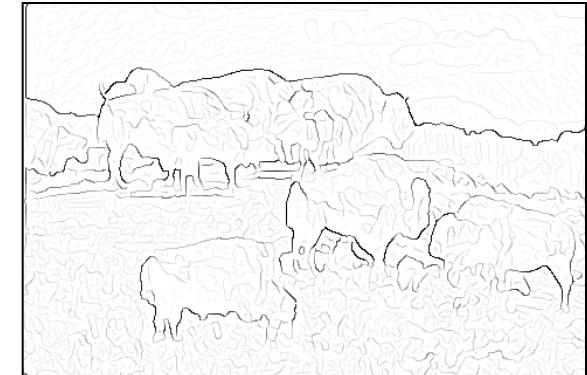
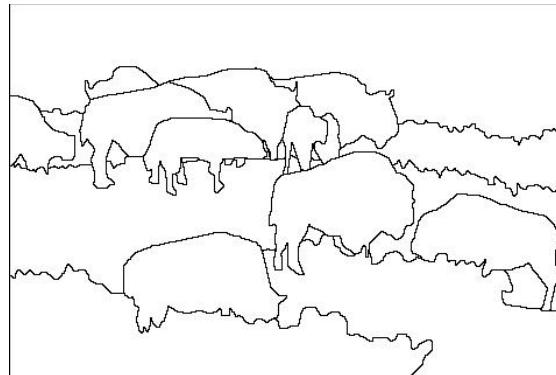
多尺度图像边缘检测



(Canny edges at multiple scales of smoothing, $\sigma=0.5, 1, 2, 4, 8, 16$)

多尺度图像边缘检测

- 面向语义的边缘检测：检测具有一定语义信息的图像边缘，例如感兴趣物体的轮廓。



人的图像边缘标定

梯度模

- Berkeley segmentation database:

<http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/>

作业

- 复习Canny边缘检测算法，用matlab命令edge,尝试不同的边缘检测算法，并比较检测结果

The Sobel method

The Prewitt method

The Roberts method

The Laplacian of Gaussian method

The zero-cross method

The Canny method

- 阅读参考文献：

Contour Detection and Hierarchical Image Segmentation, IEEE PAMI 2011

[http://www.cs.berkeley.edu/~arbelaez/publications/
amfm_pami2011.pdf](http://www.cs.berkeley.edu/~arbelaez/publications/amfm_pami2011.pdf)