



# 计算机视觉原理及实战：

## 2. 图像预处理、边缘提取及图像分割

主讲人: *Roland*



哈尔滨工业大学





图像滤波及去噪

图像边缘检测

直线检测

直方图与图像分割

图像特征描述

总结



# 图像滤波及预处理

---

- 图像滤波定义
- 图像去噪
- 图像形态学操作

➤ 与1维信号滤波类似，图像滤波由**卷积**定义：

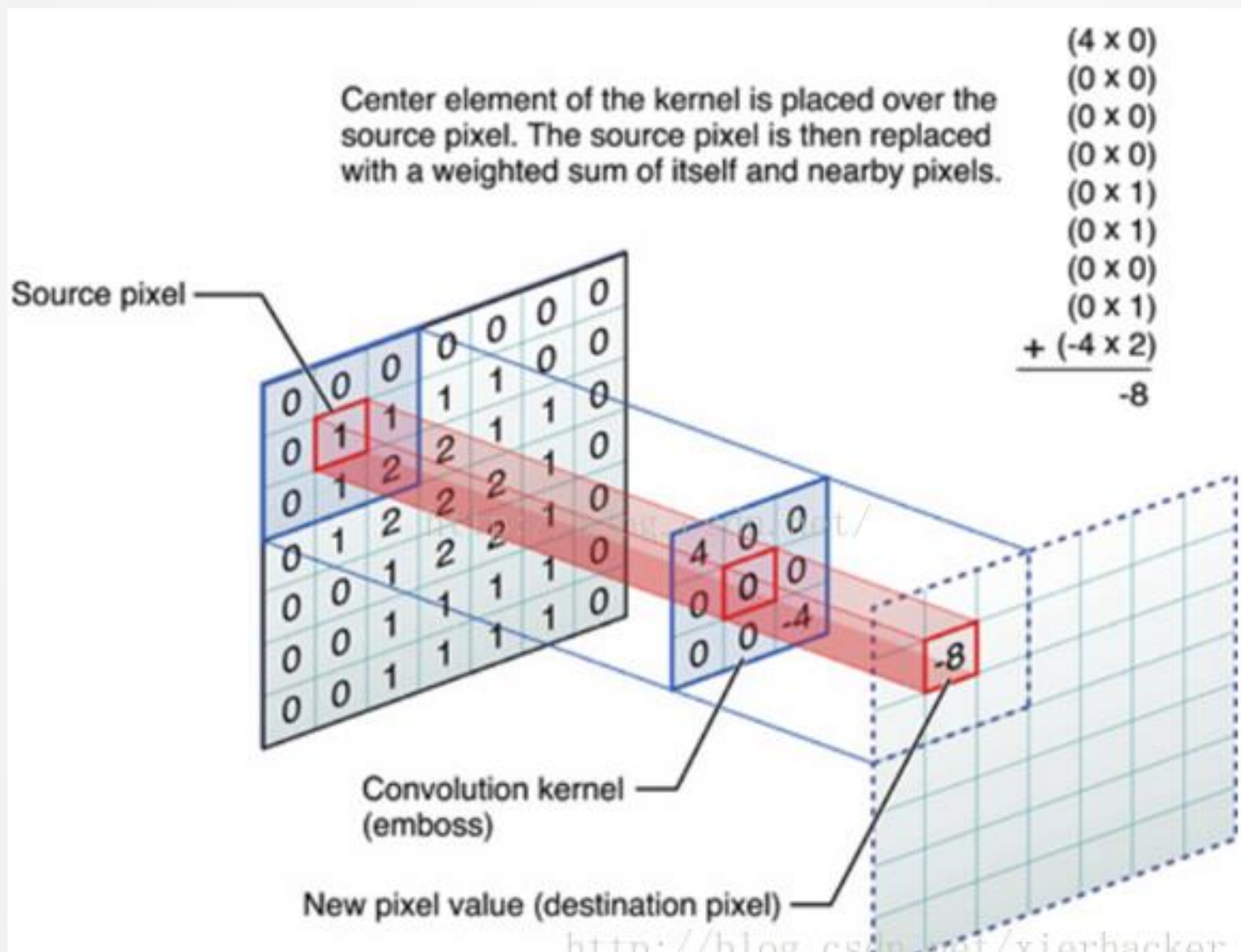
$$f(x, y) * g(x, y) = \frac{1}{NM} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} f(n, m) g(x-n, y-m)$$

➤ 在图像中，也常以**模版**的形式定义：

$$f(x, y) * g(x, y) = \frac{1}{NM} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} f(x+n, y+m) g(n, m)$$

➤ 如果滤波器对称，则二者等价

## 图像滤波如何计算——某一像素滤波结果



## 图像滤波如何计算——计算过程

1 <sub>x1</sub>	1 <sub>x0</sub>	1 <sub>x1</sub>	0	0
0 <sub>x0</sub>	1 <sub>x1</sub>	1 <sub>x0</sub>	1	0
0 <sub>x1</sub>	0 <sub>x0</sub>	1 <sub>x1</sub>	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved  
Feature

## 图像平滑——平均滤波

➤ 在一个小区域内（通常3\*3）像素值平均

$$g(x, y) = \frac{1}{M} \sum_{(m, n) \in S} f(m, n)$$

$$\frac{1}{5} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\frac{1}{8} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

## 图像平滑——加权平均滤波

➤ 在一个小区域内（通常3\*3）像素值加权平均

$$g(x, y) = \frac{1}{M} \sum_{(m,n) \in S} w_{mn} f(m, n)$$

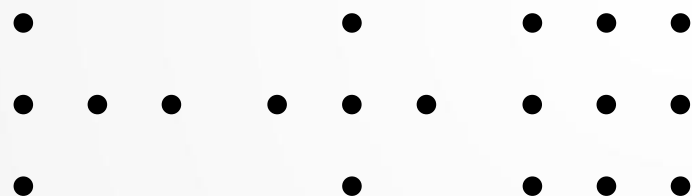
$$\frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

高斯模板

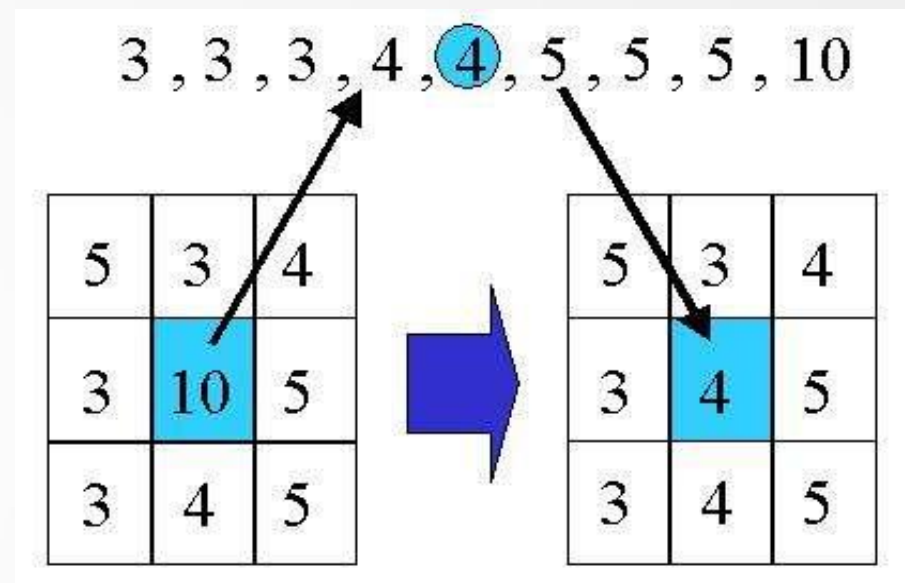
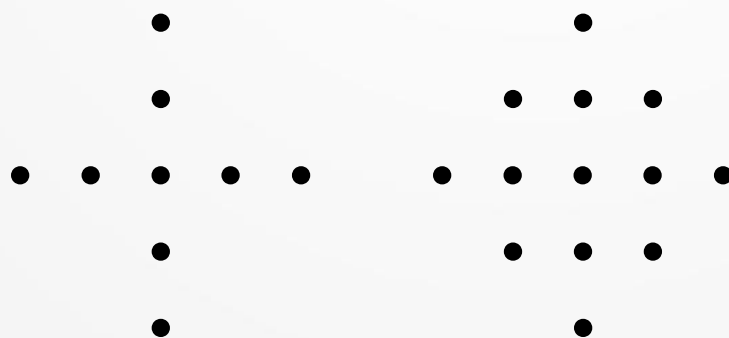


## 图像平滑——中值滤波

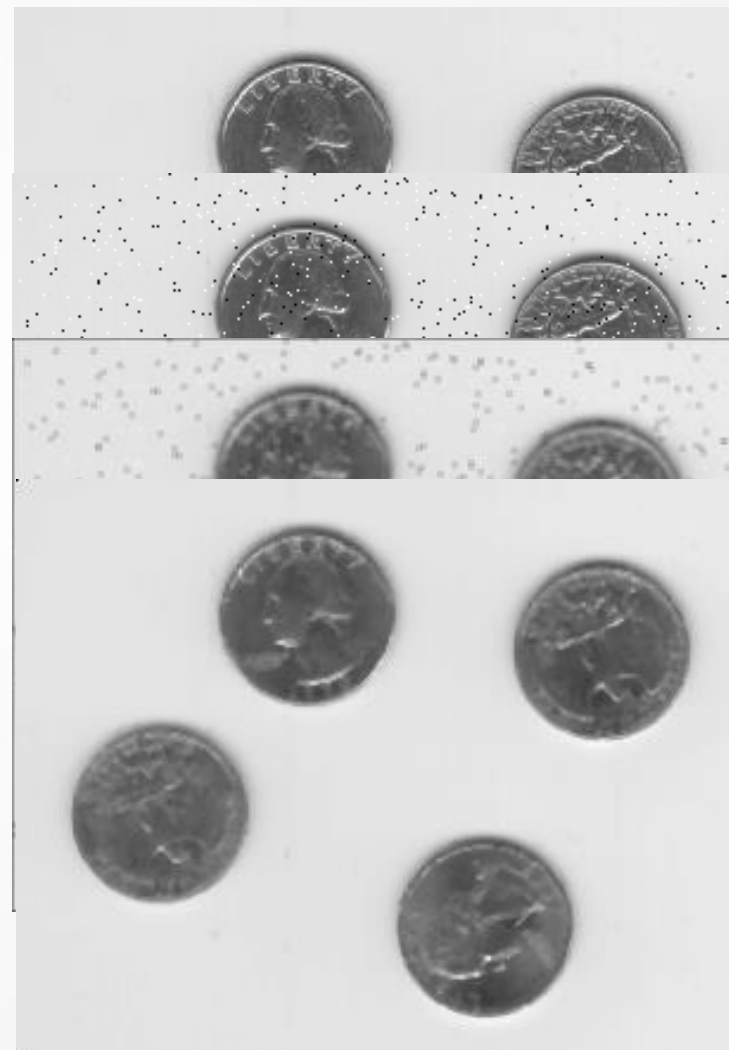
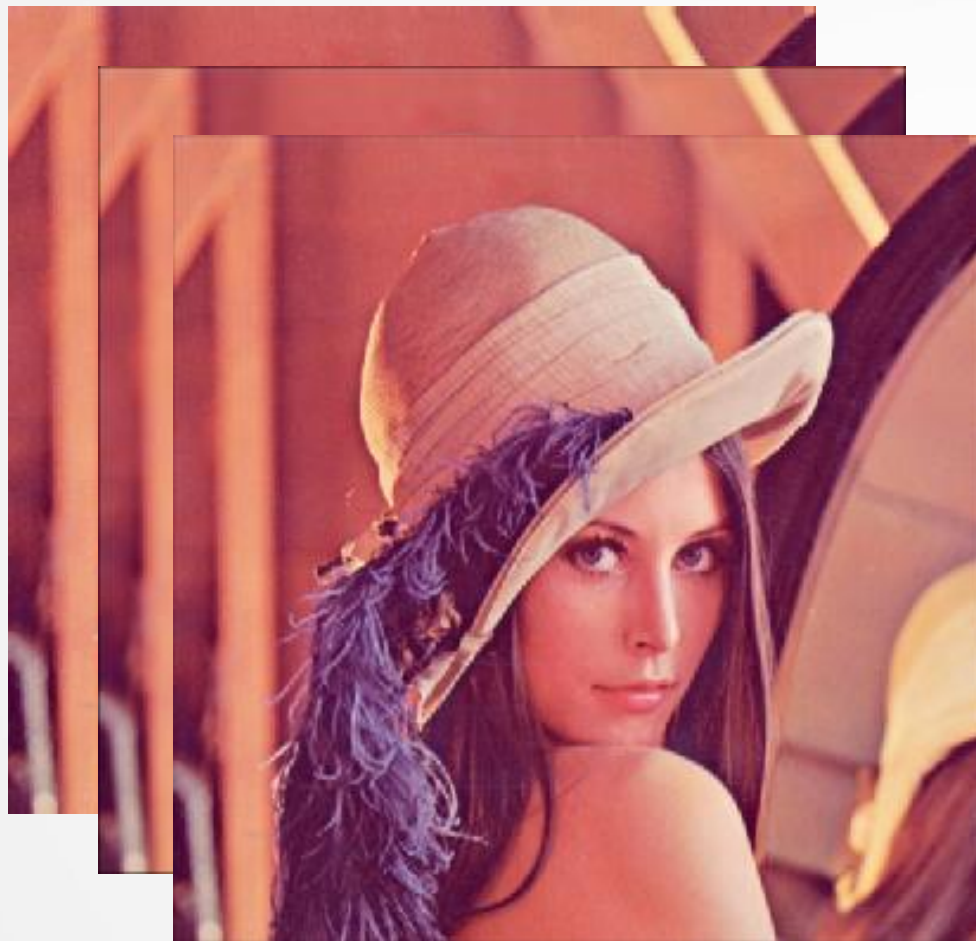
- (1) 确定窗口及位置(含有奇数个像素);
- (2) 窗口内像素按灰度大小排序;
- (3) 取中间值代替原窗口中心像素值.



对椒盐噪声有效



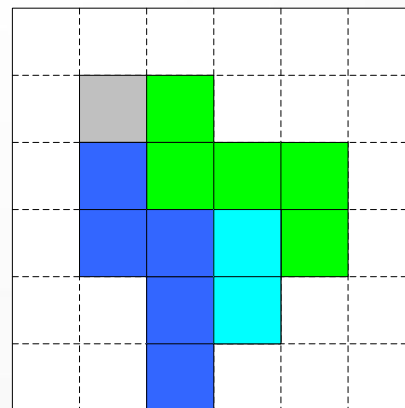
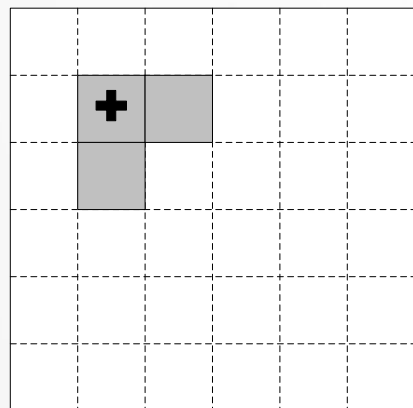
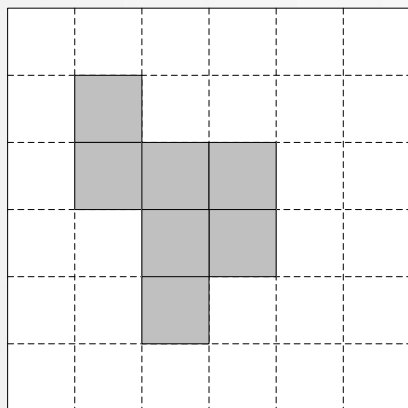
## 图像平滑——效果比较



## 图像形态学基本操作——膨胀

$A \oplus B$  表示集合  $A$  用结构元素  $B$  膨胀，定义为：

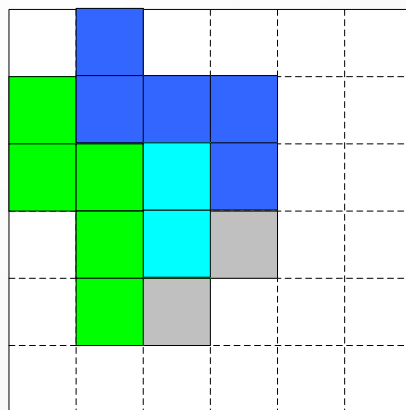
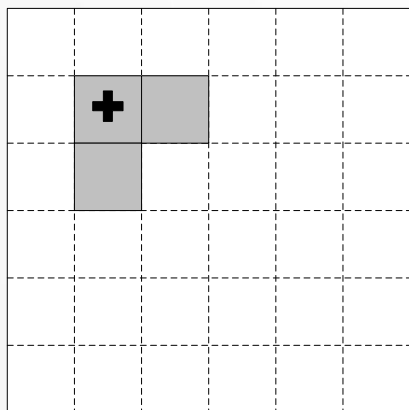
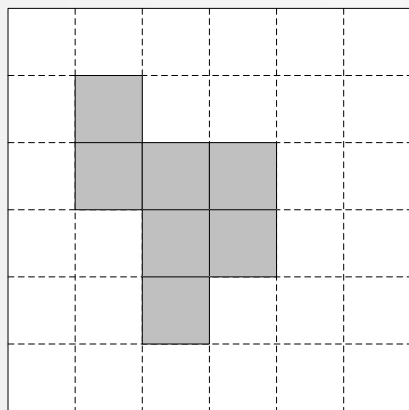
$$A \oplus B = \bigcup_{b \in B} (A)_b$$



## 图像形态学基本操作——腐蚀

$A \ominus B$  表示集合  $A$  用结构元素  $B$  腐蚀，定义为：

$$A \ominus B = \bigcap_{b \in B} (A)_{-b}$$



## 图像形态学操作——开闭运算

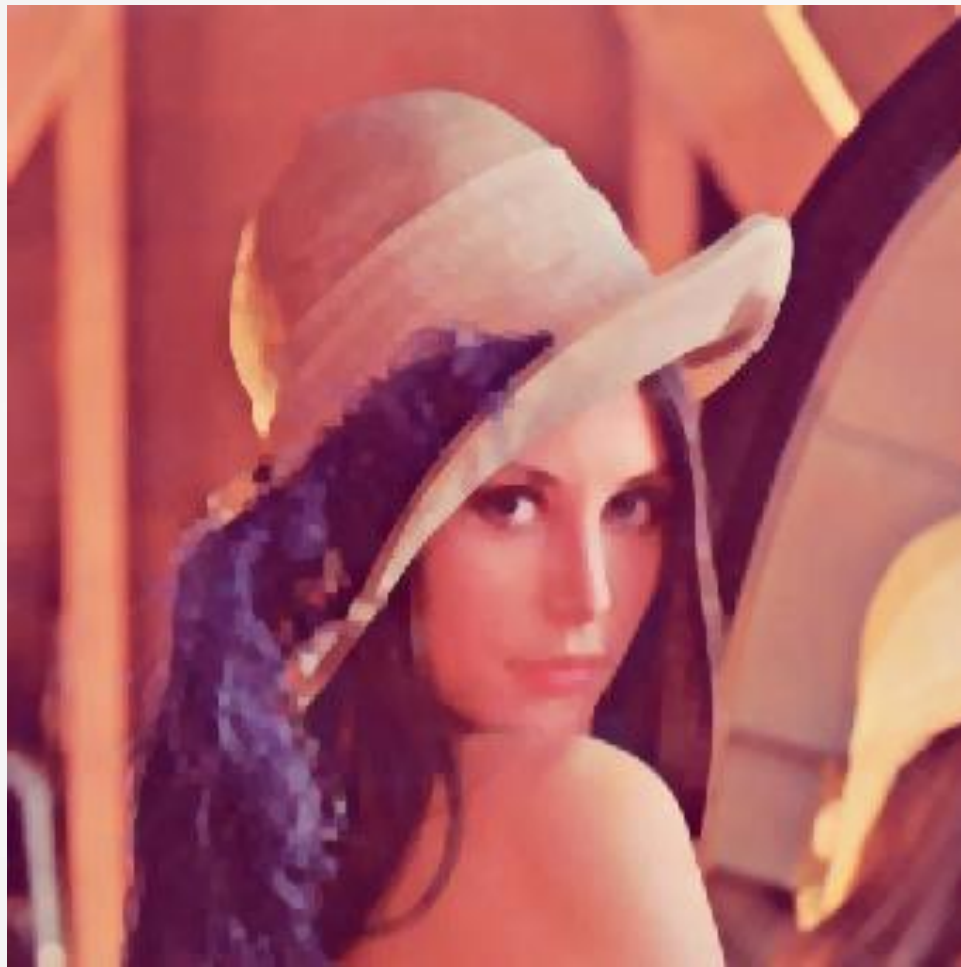
- 膨胀和腐蚀并不互为逆运算，二者级联使用可生成新的形态学运算
- 开运算：先腐蚀后膨胀；

$$A \circ B = (A \ominus B) \oplus B$$

- 闭运算：先膨胀后腐蚀

$$A \bullet B = (A \oplus B) \ominus B$$

- 先开后闭：可有效去除噪声





# 图像边缘检测

---

- 基本边缘检测算子
- Canny算子

## 图像边缘检测——基本算子

Sobel 算子卷积核

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

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

Prewitt 算子卷积核

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

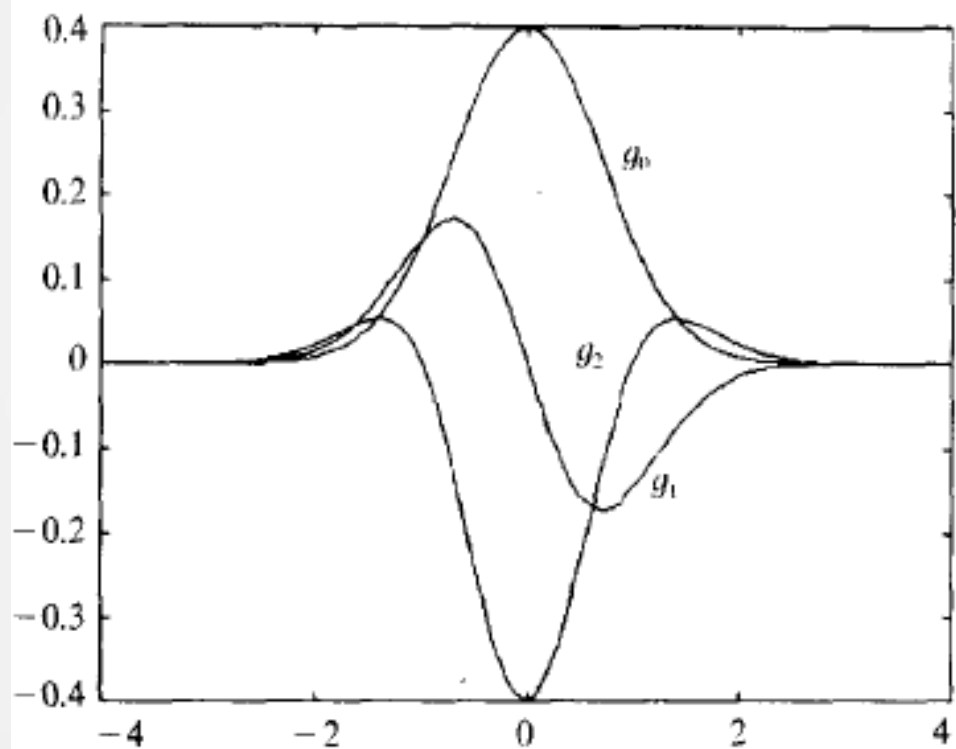
1	0	-1
1	0	-1
1	0	-1

- 边缘检测的本质是微分
- 实际中常用差分， $x$ 方向和 $y$ 方向



## 图像边缘检测——Laplace算子

$$\Delta[G_{\sigma}(x, y) * f(x, y)] = [\Delta G_{\sigma}(x, y)] * f(x, y) = LoG * f(x, y)$$



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

已有方法问题：噪声；断裂；虚检(渐变灰度)

Canny算子核心优点：边缘可自动连通

算法步骤：

1. 平滑图像同时计算微分；
2. 计算梯度(幅值和方向)；
3. 梯度幅值进行非极大值抑制；
4. 自动边缘连接

## Canny算子计算步骤1：同时平滑与微分

➤ 使用高斯函数的一阶导数同时完成平滑和微分

$$H(x, y) = \nabla (G(x, y) * f(x, y)) = \nabla G(x, y) * f(x, y)$$

$$\nabla G = \begin{bmatrix} \partial G / \partial x \\ \partial G / \partial y \end{bmatrix}$$

$$G(x, y) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{x^2 + y^2}{\sigma^2}\right]$$

## Canny算子计算步骤2：梯度计算

$$E_x = \frac{\partial G}{\partial x} * f(x, y) \quad E_y = \frac{\partial G}{\partial y} * f(x, y)$$

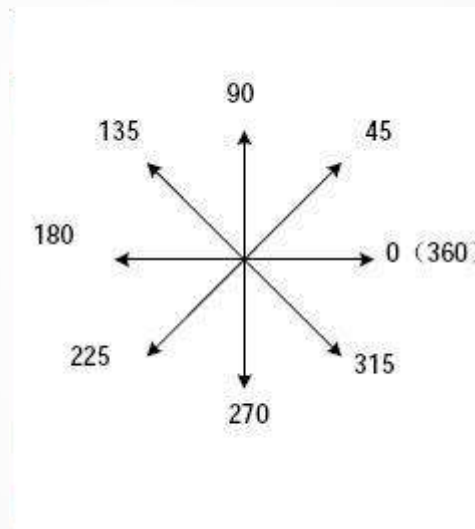
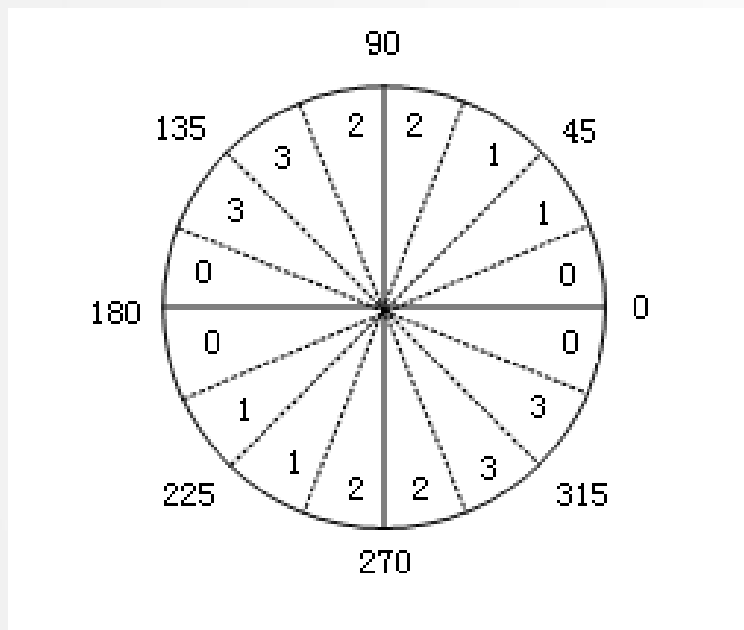
➤ 进一步可得梯度幅值和方向：

$$M(i, j) = \sqrt{E_x^2(i, j) + E_y^2(i, j)}$$

$$\theta(i, j) = \arctg \left[ \frac{E_x(i, j)}{E_y(i, j)} \right]$$

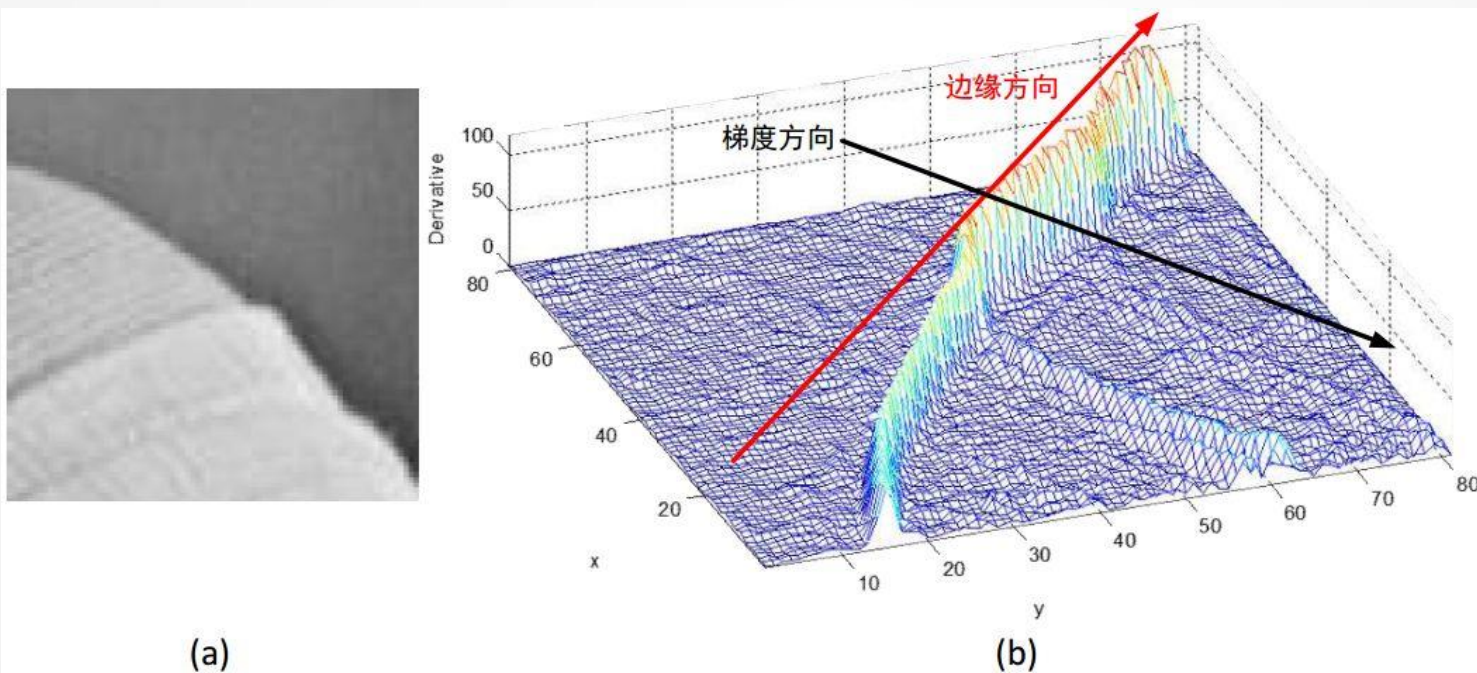
## Canny算子计算步骤2：梯度计算

➤ 方向离散化：离散化为上下左右和斜45° 共4个方向



## Canny算子计算步骤3：梯度幅值非极大值抑

- 细化梯度幅值图像中的屋脊带，只保留幅值局部变化最大的点
- 使用一个 $3 \times 3$ 邻域作用于幅值阵列的所有点。在每一点上，邻域的中心像素与沿梯度方向的两个梯度幅值的插值结果进行较，仅保留极大值点。



## Canny算子计算步骤4：边缘连接

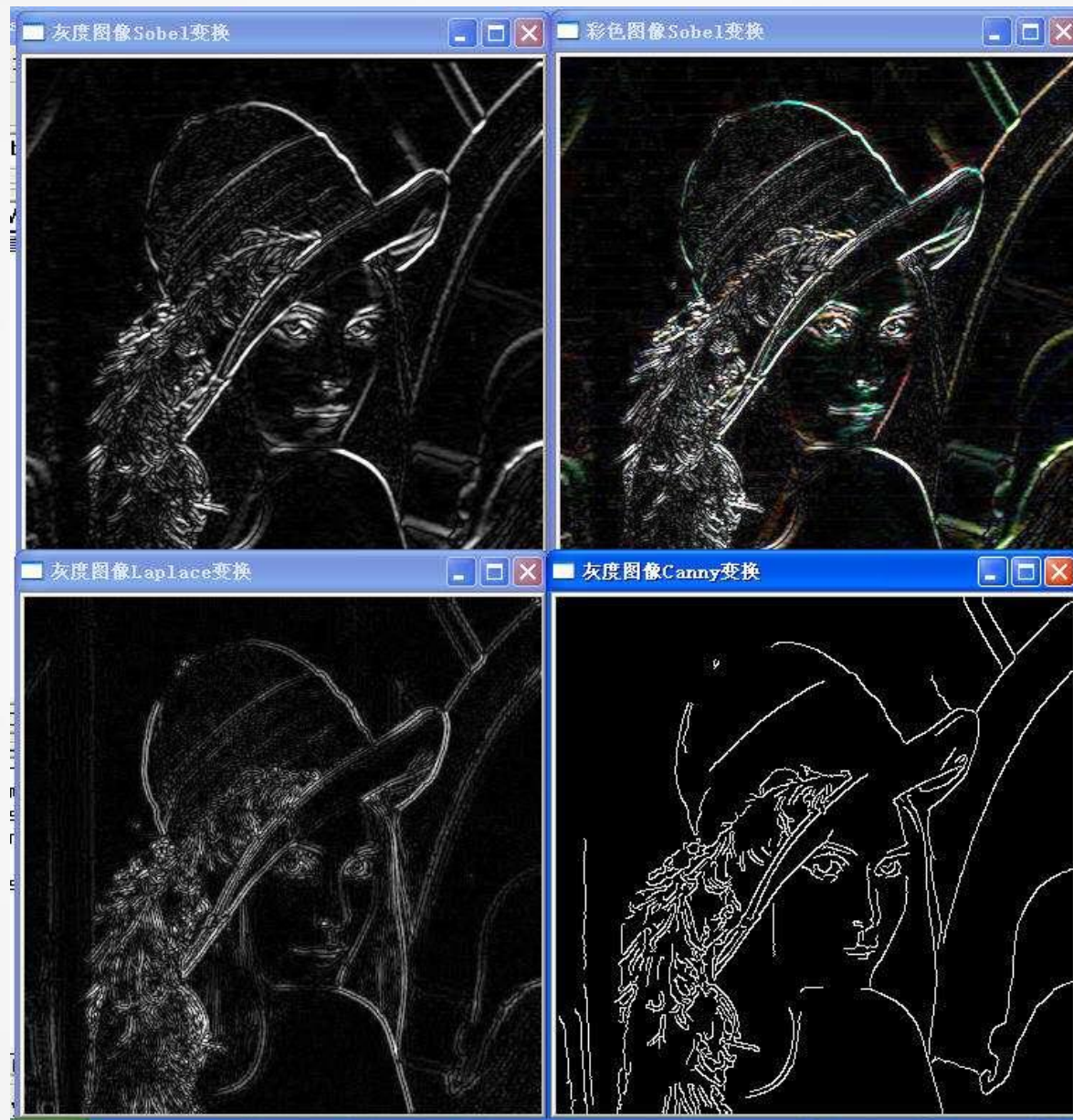
➤ 对上一步得到的图像使用低、高阈值  $\tau_1$   $\tau_2$  阈值化，得到三幅图像：

$$\begin{cases} T_1[i, j](M(i, j) < \tau_1) \\ T_2[i, j](\tau_1 \leq M(i, j) \leq \tau_2) \\ T_3[i, j](M(i, j) > \tau_2) \end{cases}$$

- $T_1$  对应假边缘，去除；
- $T_3$  对应真边缘，全部保留；
- $T_2$  连接：临接像素中是否有属于  $T_3$  的像素



边缘检测  
算子结果  
比较





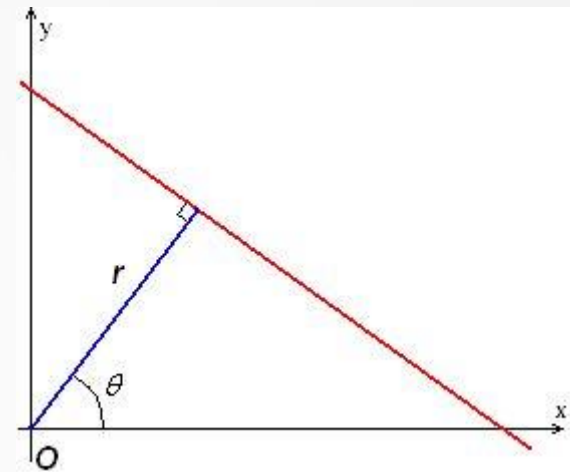


# 直线检测

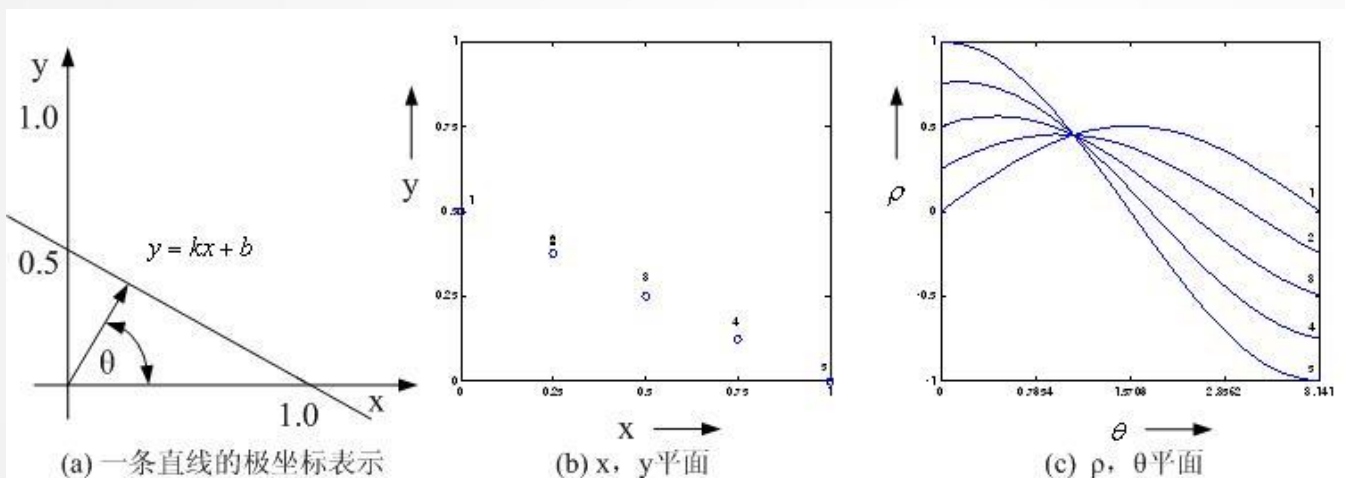
---

- 光通量
- 辐照度

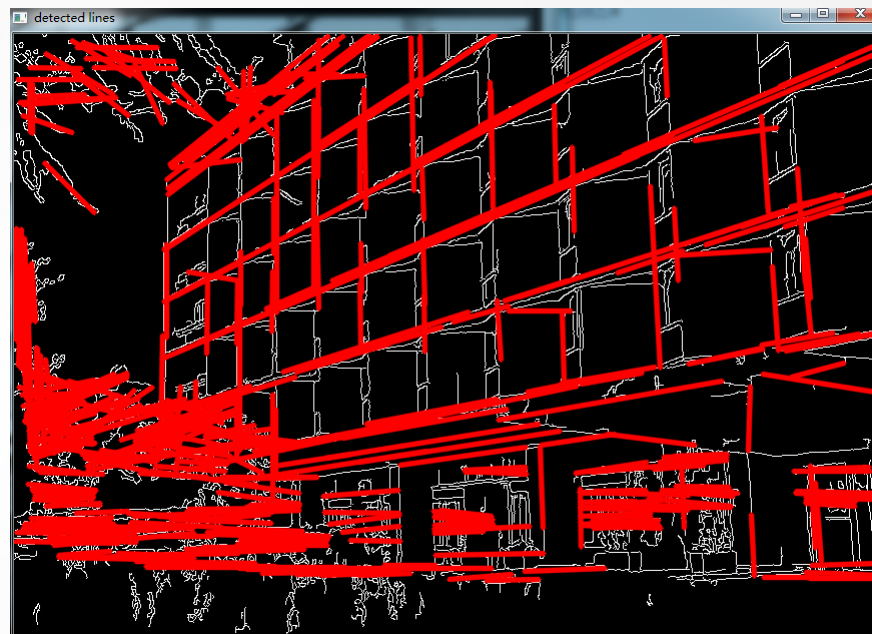
- 采用参数空间变换的方法，对噪声和不间断直线的检测具有鲁棒性
- 可用于检测圆和其他参数形状
- 核心思想：直线  $y = kx + b$   
每一条直线对应一个  $k, b$ ，极坐标下对应一个点  $(\rho, \theta)$



- 直角坐标系的一点 $(x,y)$ ，对应极坐标系下的一条正弦曲线  $\rho = x \cos \theta + y \sin \theta$
- 同一条直线上的多个点，在极坐标系下必相交于一点



- 将 $(\rho, \theta)$ 空间量化成许多小格;
- 根据 $x$ - $y$ 平面每一个直线点代入 $\theta$ 的量化值, 算出各个 $\rho$ , 将对应格计数累加;
- 当全部点变换后, 对小格进行检验。设置累计阈值 $T$ , 计数器大于 $T$ 的小格对应于共线点, 其可以用作直线拟合参数。小于 $T$ 的反映非共线点, 丢弃不用。

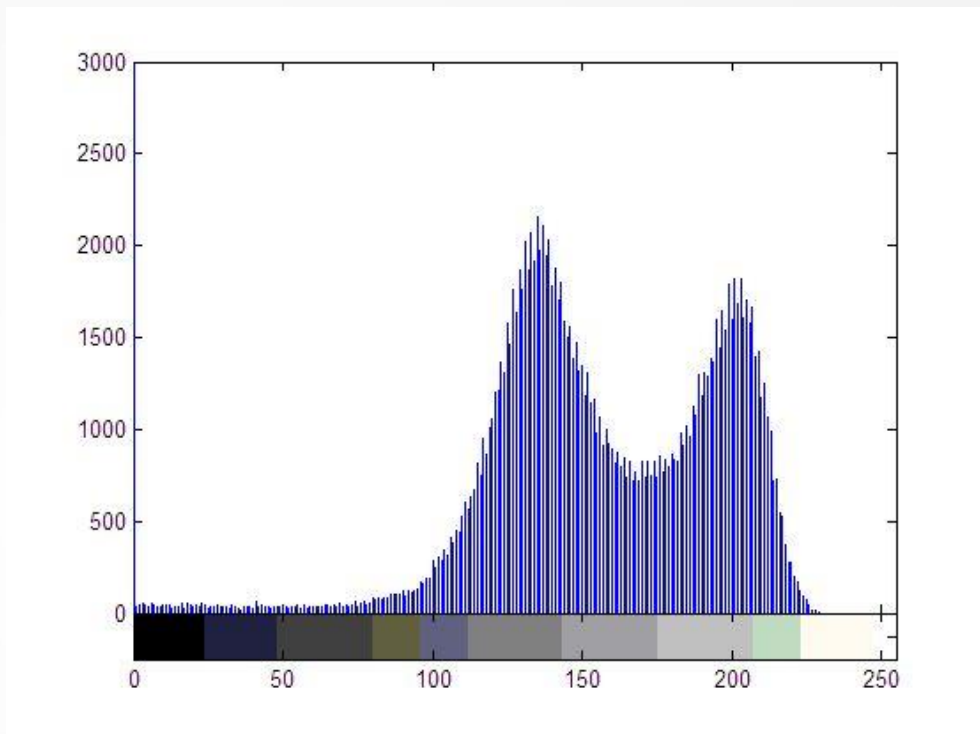
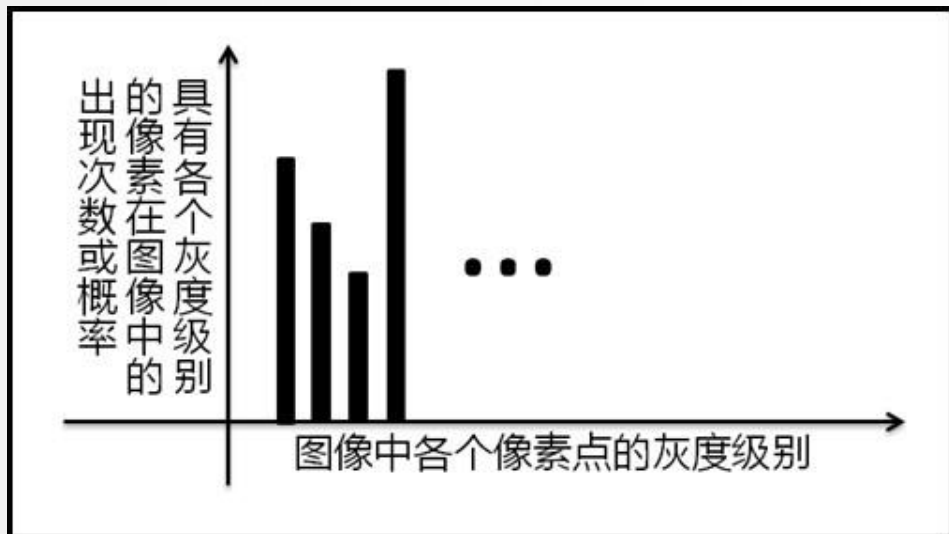




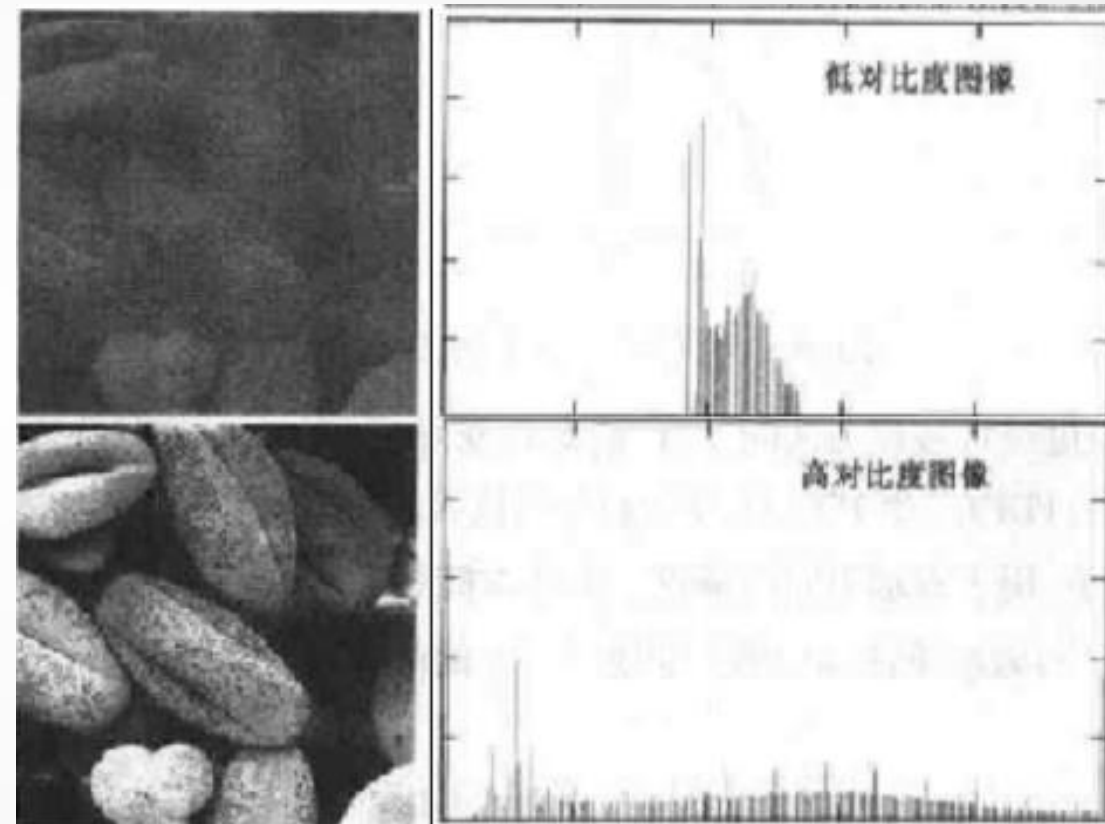
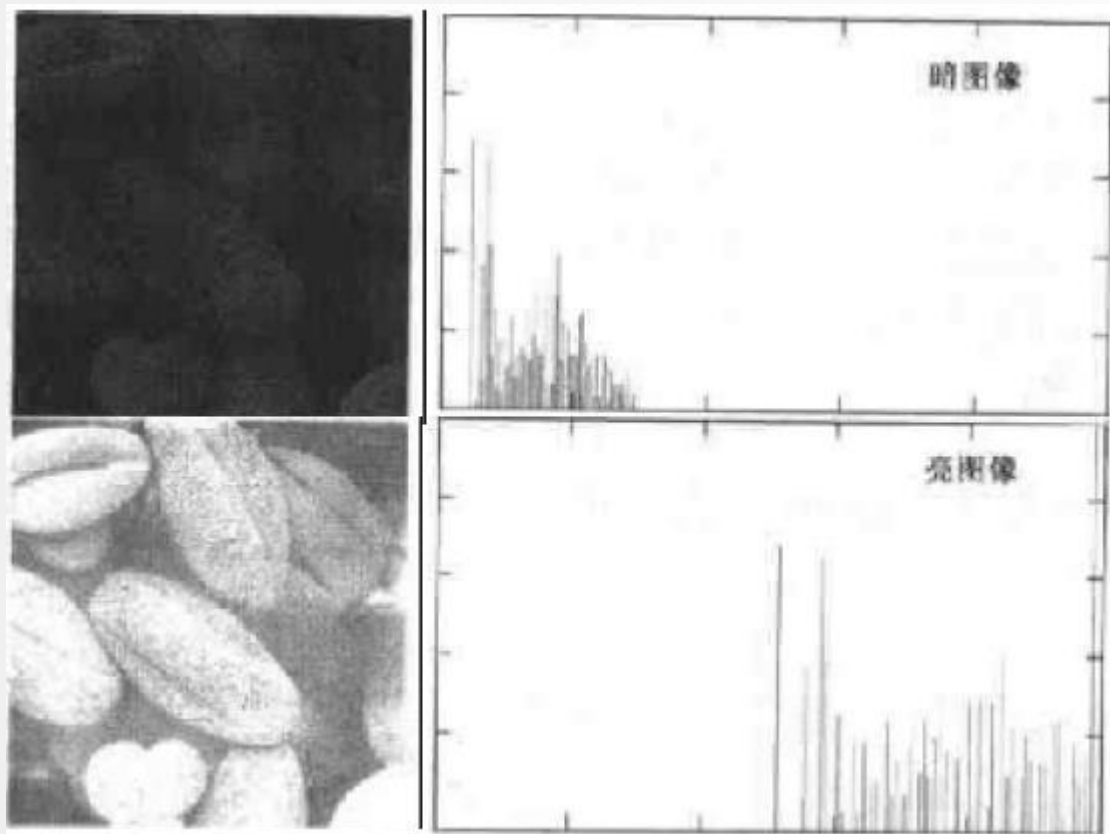
# 直方图与阈值分割

---

- 图像直方图
- 灰度阈值分割



## 通过灰度直方图看到图像的照明效果

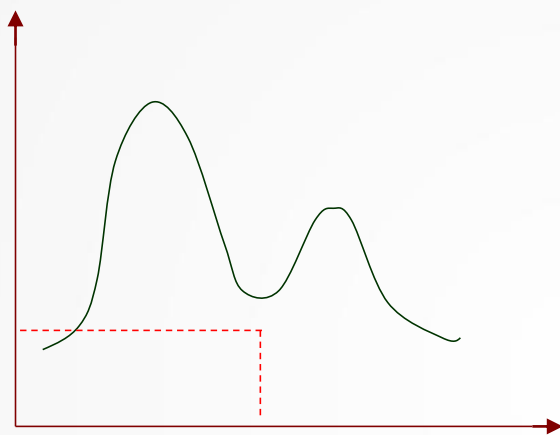




- 假设：图像中的目标区和背景区之间或者不同目标区之间，存在不同的灰度或平均灰度
- 凡是灰度值包含于 $z$ 的像素都变成某一灰度值，其他的变成另一个灰度值，则该图像就以 $z$ 为界被分成两个区域

$$f(x, y) = \begin{cases} z_a & f(x, y) \in z \\ z_b & \text{其他} \end{cases}$$


- 如果 $z=1$ 和 $z=0$ ，分割后的图像为二值图像



➤ 确定最佳阈值，使背景和目标之间的差异最大

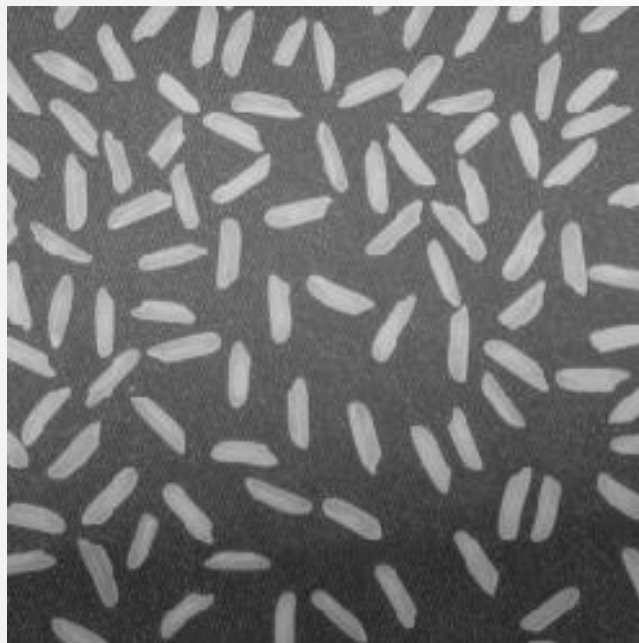
## 大津(Otsu)算法

- 确定最佳阈值，使背景和目标之间的类间方差最大（因为二者差异最大）

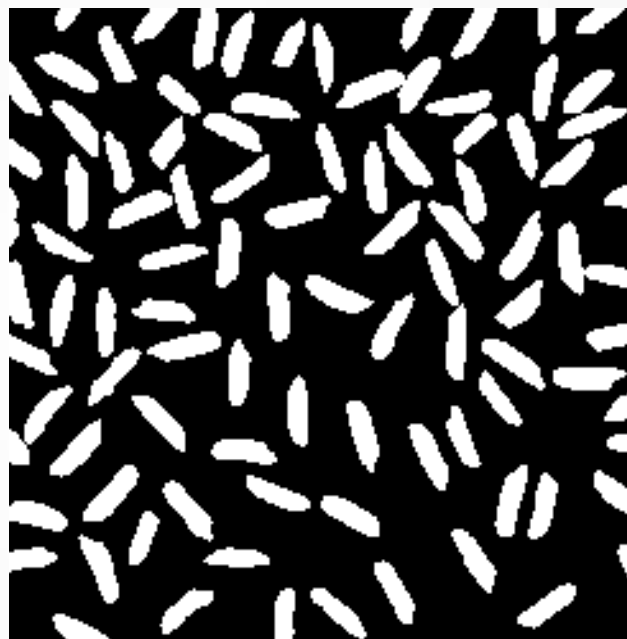

$$\begin{aligned}\mu &= \omega_0\mu_0 + \omega_1\mu_1 \\ g &= \omega_0(\mu_0 - \mu)^2 + \omega_1(\mu_1 - \mu)^2 \\ g &= \omega_0\omega_1(\mu_0 - \mu_1)^2\end{aligned}$$

- 算法实现：遍历灰度取值

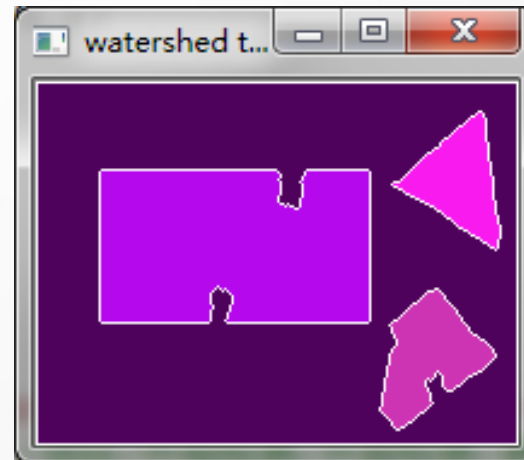
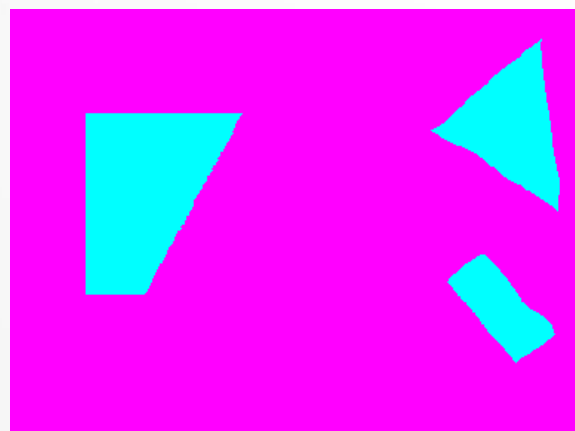
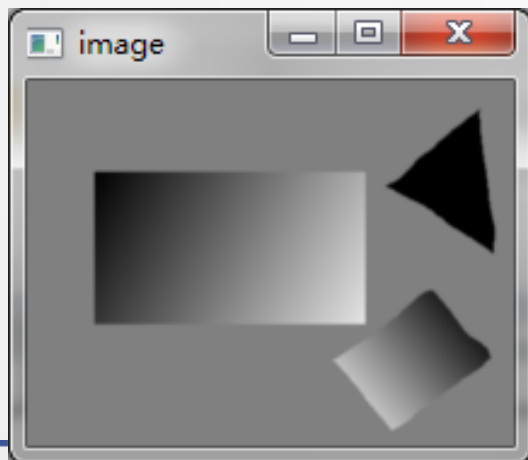
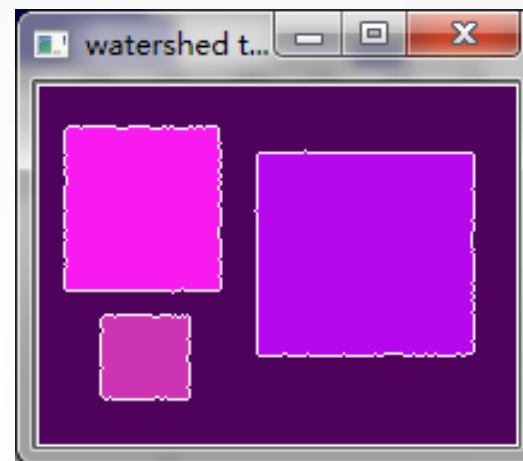
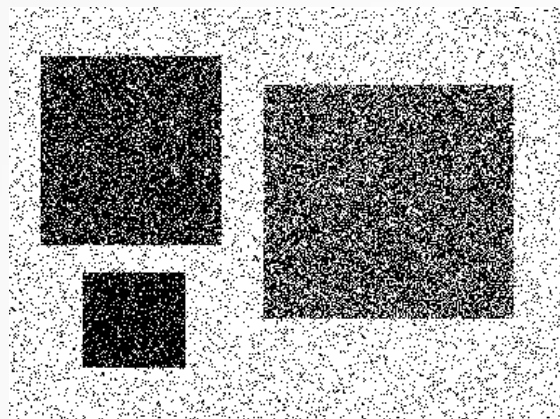
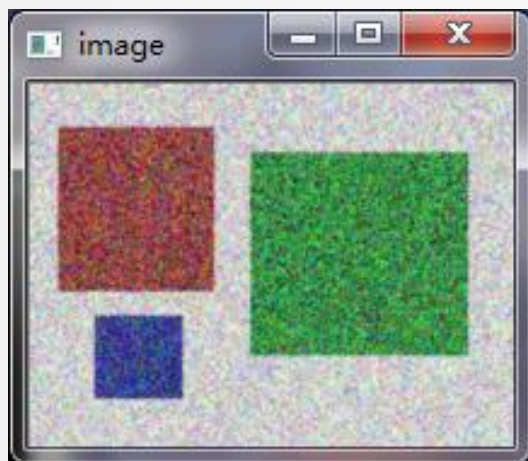
原始米粒图像



使用Otsu算法进行阈值化结果



## 大津算法局限性及其它方法





# 图像特征描述

---

- 简单描述
- 一般性描述

## ➤ 简单描述符

- 区域面积：区域包含的像素数
- 区域重心：

$$x = \frac{1}{A} \sum_{(x,y) \in R} x$$
$$y = \frac{1}{A} \sum_{(x,y) \in R} y$$

区域重心可能不是整数

## ➤ 形状描述符

➤ 形状参数:  $F = \frac{\|B\|^2}{4\pi A}$  圆形  $F=1$ ;  
其他  $F>1$

➤ 偏心率: 等效椭圆宽高比

➤ 欧拉数:  $E=C-H$

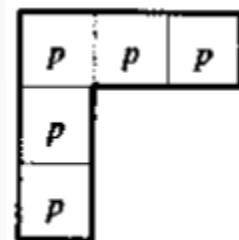
➤ 圆形性:  $C = \frac{\mu_R}{\sigma_R}$

$$\mu = \frac{1}{K} \sum_{(x_i, y_i) \in E(R)} \|(x_i, y_i) - (\bar{x}, \bar{y})\|$$
$$\sigma = \frac{1}{K} \sum_{(x_i, y_i) \in E(R)} [\|(x_i, y_i) - (\bar{x}, \bar{y})\| - \mu]^2$$

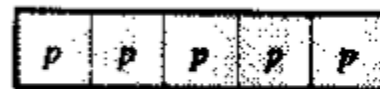


## ➤ 形状描述符实例

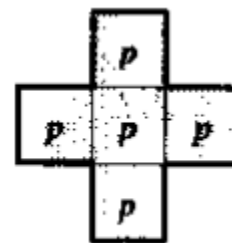
Bird



F1

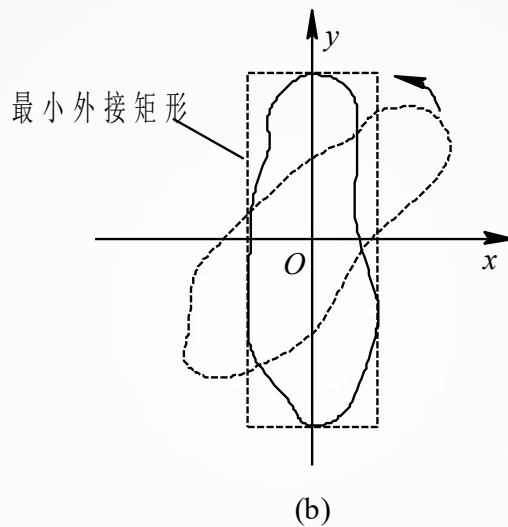
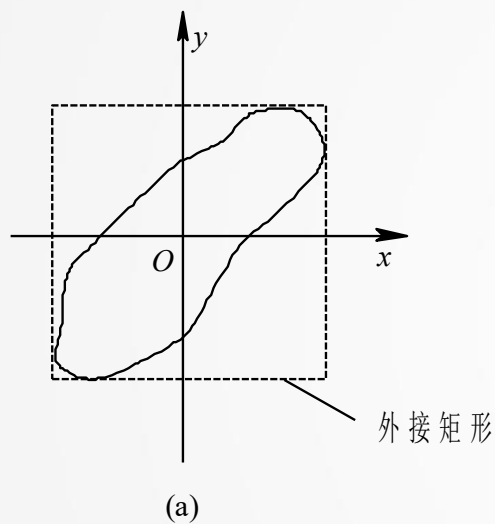


F2



F3

➤ 最小包围矩形 (MER)

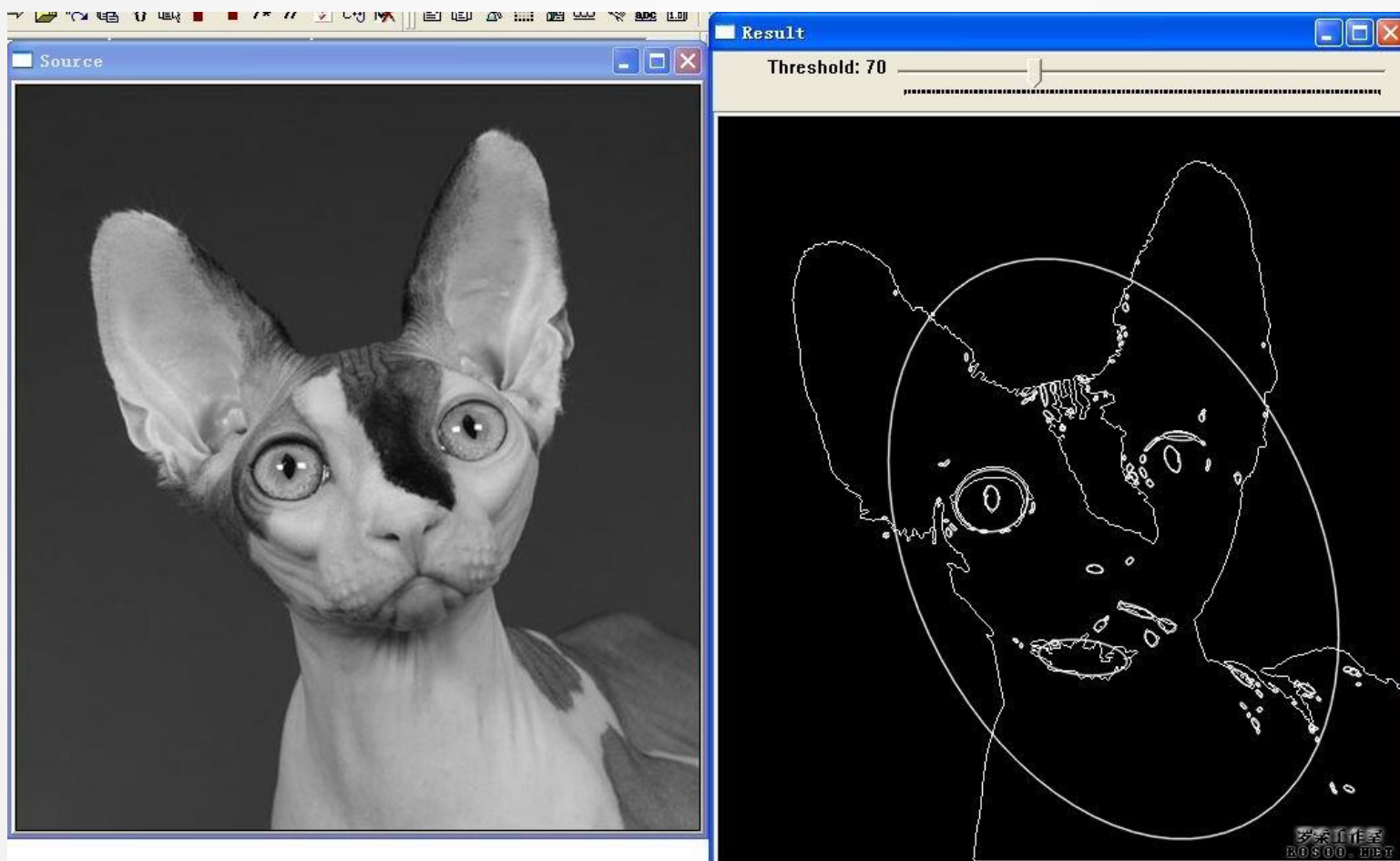


➤ 方向和离心率

➤ 椭圆拟合

$$E = \sqrt{1 - \frac{a^2}{b^2}}$$

➤ 椭圆拟合演示





# 总结

---

- 平滑滤波：常用的是高斯滤波和中值滤波
- 边缘检测：**Sobel**简单实用，**Canny**算子效果好
- 直线检测：**Hough**变换，核心思想是参数变换和投票法
- 阈值分割：大津法是最常用的分割方式之一
- 图像描述：可描述区域的形状、面积等参数



# 感谢各位的聆听！

Roland

PPT版权属于作者，PPT中引用的图像，网页等版权属于各自持有者



AI100