

# 第五章：图像表达-关键点检测

---

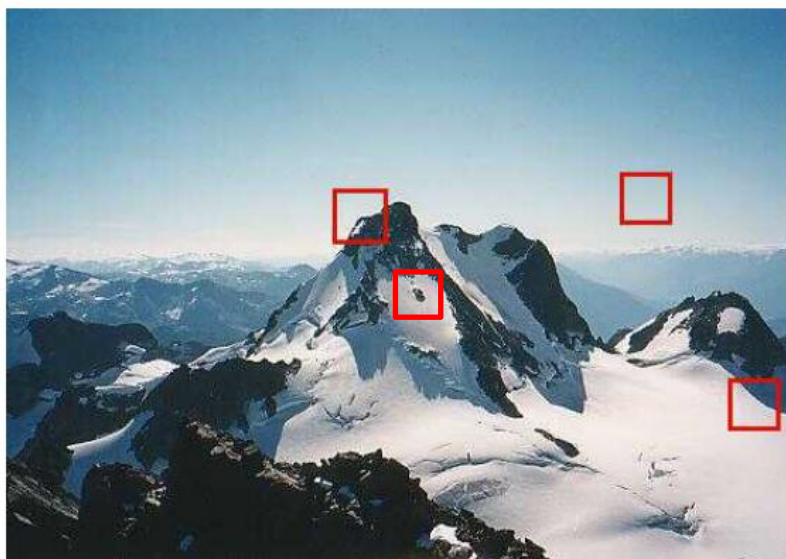
授课老师：李厚强，胡洋，周文罡，李礼

# 图像表达-关键点检测：动机

□ 如何在两幅相关的图像构建局部的关联关系？

■ 目标：在图像之间建立明确的、无二义性的匹配

✓ 如何定义这种“与众不同”？



平滑区域

(×)



边缘

(×)



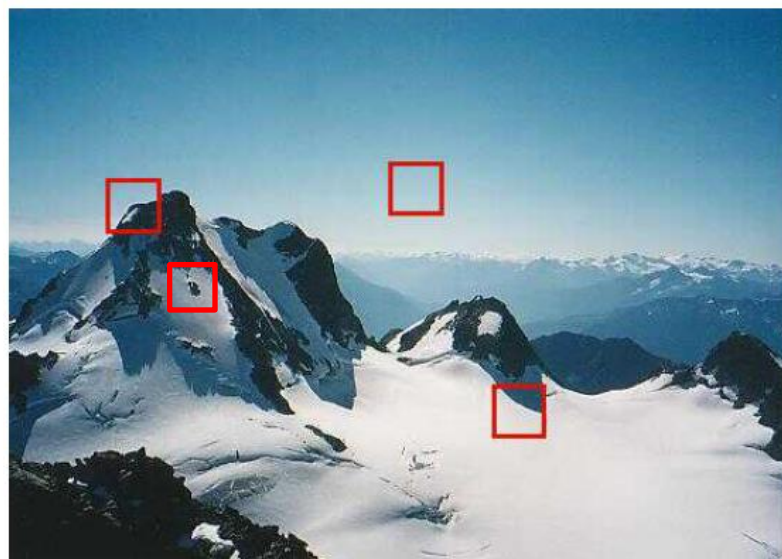
角点

(✓)



块(blob)

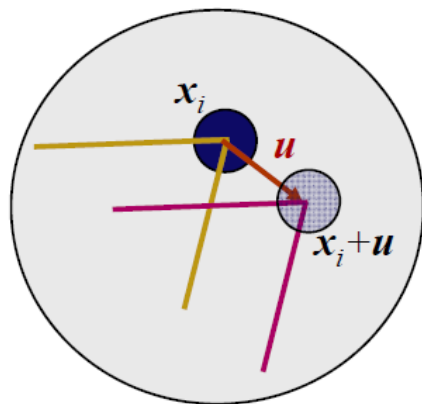
(✓)



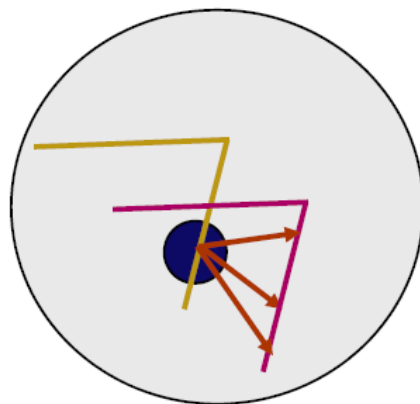
# 关键点检测：分析

## □ 孔径(aperture)问题

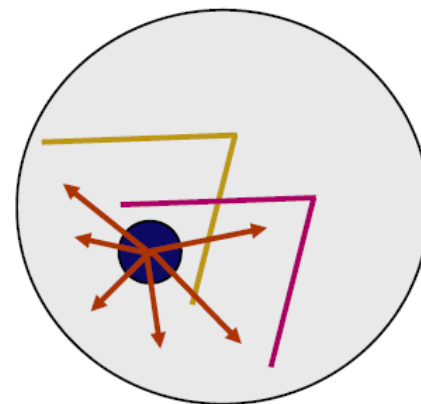
- 左(a): 图像块(patch)有两个显著的梯度方向, 容易关联
  - ✓ 角点(corner), 块(blob)
- 中(b): 直线上的图像块(patch)只能沿垂直于边缘的方向对齐
- 右(c): 无纹理(textureless)的图像块几乎无法对齐关联



(a)



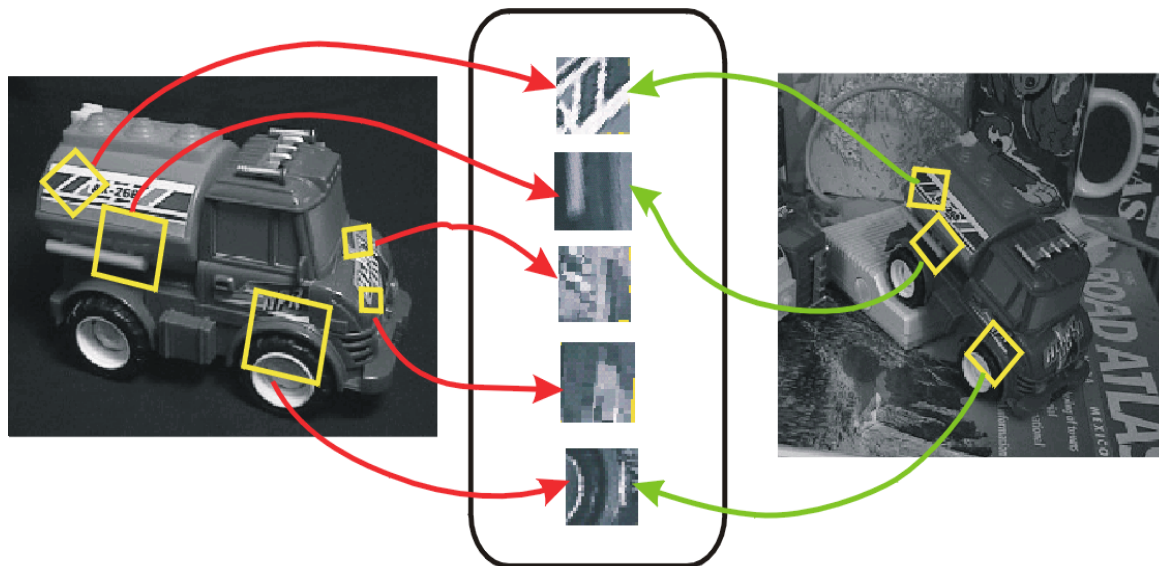
(b)



(c)

# 关键点检测

- 关键点的描述：位置 and 对应局部区域的尺寸
  - Key point, interest point
- 关键点的不变性 (invariance)
  - 以不变应万变：图像在像素空间发生了亮度变换或几何变换，但是在特征空间没有变化，或变化很小
  - 常见的不变性
    - ✓ 颜色不变性：亮度不变性，
    - ✓ 几何不变性：平移不变性，旋转不变性，尺度（缩放）不变性



# 关键点检测

## □ 关键点的基本性质

### ■ 可重复性 (Repeatability)

- ✓ 对每副图像，局部关键点检测独立进行，类似“管中窥豹”
- ✓ 图像经过变换后，对应的局部特征仍然存在

### ■ 紧凑性和高效性 (Compactness and efficiency)

- ✓ 特征数目远少于图像像素数

### ■ 局部性 (Locality)

- ✓ 每个特征对应着一个相对较小的图像区域;
- ✓ 对于混乱背景和遮挡具有鲁棒性



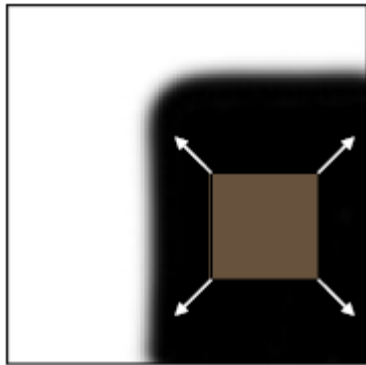


# 图像表达-关键点检测

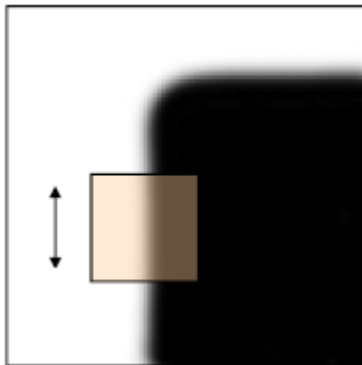
- 角点检测 (Corner Detection)
  - **Harris检测子**
  - FAST检测子
- 块检测 (Blob Detection)
  - Laplacian-of-Gaussian (LoG) 检测子
  - Difference-of-Gaussian (DoG)检测子
  - SURF检测子
  - MSER检测子

# 关键点检测：角点

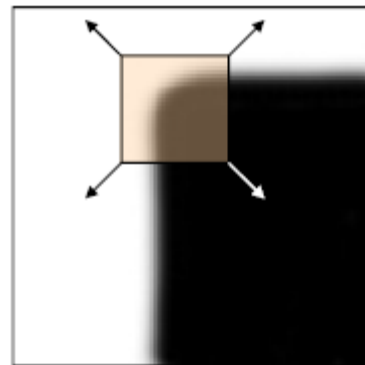
- 通过查看一个小窗口，即可简单的识别角点
  - “管中窥豹”
- 在角点上，向任何一个方向移动窗口，都会产生灰度的较大变化



“平坦”区域：  
任意方向灰度  
均无变化



“边缘”：沿边  
缘方向，灰度无  
变化



“角点”：所有  
方向上，灰度均  
有较大变化

# Harris 角点检测

□ 考虑一个滑窗 $W$ ，平移量为 $(u, v)$

- 滑动前后，滑窗 $W$ 中的像素灰度如何变化？
- 比较滑窗内对应像素灰度，计算平方差之和

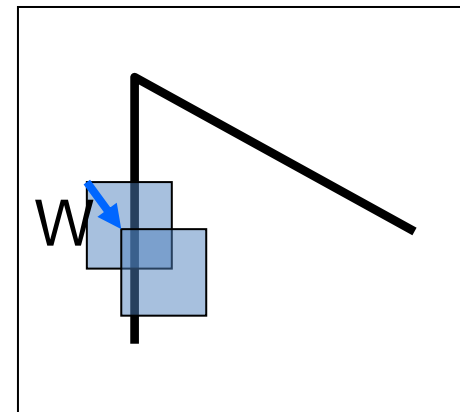
$$E(u, v) = \sum_{(x, y) \in W} [I(x + u, y + v) - I(x, y)]^2$$

- 对 $I$ 关于 $x$ 、 $y$ 做泰勒展开：

$$I(x + u, y + v) = I(x, y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \text{higher order terms}$$

- 如果滑窗平移量 $(u, v)$ 足够小，则可去掉高阶项：

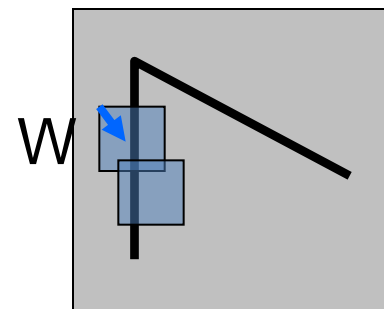
$$\begin{aligned} I(x + u, y + v) &\approx I(x, y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v \\ &\approx I(x, y) + [I_x \ I_y] \begin{bmatrix} u \\ v \end{bmatrix} \end{aligned}$$



# Harris 角点检测

$$E(u, v) = \sum_{(x,y) \in W} [I(x + u, y + v) - I(x, y)]^2$$

$$I(x + u, y + v) \approx I(x, y) + [I_x \ I_y] \begin{bmatrix} u \\ v \end{bmatrix}$$

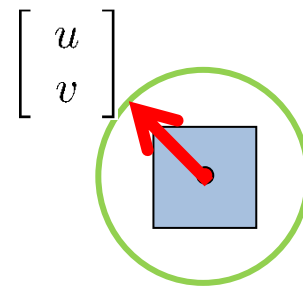
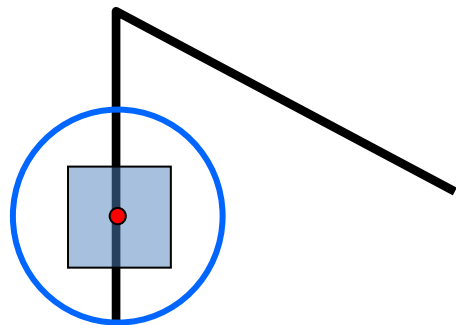


$$\begin{aligned}
 E(u, v) &= \sum_{(x,y) \in W} [I(x + u, y + v) - I(x, y)]^2 \\
 &\approx \sum_{(x,y) \in W} [I(x, y) + [I_x \ I_y] \begin{bmatrix} u \\ v \end{bmatrix} - I(x, y)]^2 \\
 &\approx \sum_{(x,y) \in W} \left[ [I_x \ I_y] \begin{bmatrix} u \\ v \end{bmatrix} \right]^2
 \end{aligned}$$

# Harris 角点检测

□ 上式可重写为：

$$\square E(u, v) = [u \quad v] \sum_{(x,y) \in W} \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = [u \quad v] \mathbf{M} \begin{bmatrix} u \\ v \end{bmatrix}$$



□ 对于上方的例子

- 可以将蓝色滑窗的中心移到移到绿色单位圆上的任意位置
- 朝那个方向 $[u \quad v]$ 移动，会得到最大和最小的 $E$  值呢？
  - ✓ 可通过对矩阵 $\mathbf{M}$  的特征向量，得到这两个方向



# 回顾：特征向量、特征值

□ 对于矩阵 $\mathbf{A}$ ，其特征向量为 $\mathbf{x}$ ，当其满足：

$$\mathbf{Ax} = \lambda\mathbf{x}$$

■ 标量 $\lambda$ 为特征向量 $\mathbf{x}$ 所对应的特征值

□ 特征值可通解如下方程得到： $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$

□ 在上页PPT上， $\mathbf{A} = \mathbf{M}$ 是一个 $2 \times 2$ 的矩阵，因此可得

$$\det \begin{pmatrix} m_{11} - \lambda & m_{12} \\ m_{21} & m_{22} - \lambda \end{pmatrix} = 0$$

■ 其解为：

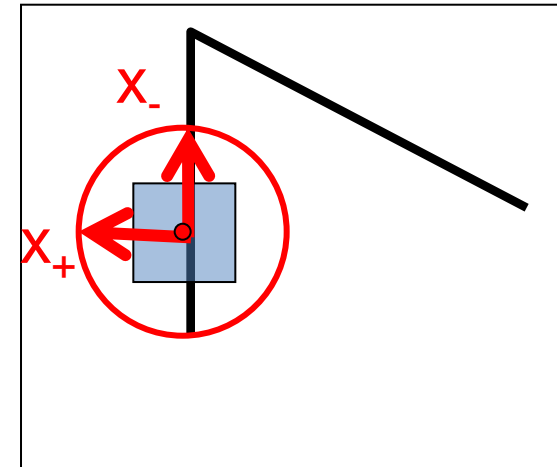
$$\lambda_{\pm} = \frac{1}{2} \left[ (m_{11} + m_{22}) \pm \sqrt{4m_{12}m_{21} + (m_{11} - m_{22})^2} \right]$$

□ 一旦得到特征值 $\lambda$ ，可以通过解如下方程得到特征向量 $\mathbf{x}$

$$\begin{bmatrix} m_{11} - \lambda & m_{12} \\ m_{21} & m_{22} - \lambda \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0$$

# Harris 角点检测

$$E(u, v) = [u \ v] \underbrace{\sum_{(x,y) \in W} \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix}}_M \begin{bmatrix} u \\ v \end{bmatrix}$$



## □ 矩阵M 的特征值和特征向量

- Define shifts with the largest and smallest change (E value)
- $x_+$  = direction of largest increase in E.
- $\lambda_+$  = amount of increase in direction  $x_+$

$$Mx_+ = \lambda_+ x_+$$

- $x_-$  = direction of smallest increase in E.
- $\lambda_-$  = amount of increase in direction  $x_+$

$$Mx_- = \lambda_- x_-$$



# Harris 角点检测

## □ 滑窗中的灰度变化：特征值分析

- $\lambda_{\min}, \lambda_{\max}$  : 对称矩阵  $\mathbf{M}$  的特征值
- 如果尝试所有可能的平移方向向量  $\mathbf{n}$  , 那么产生最大灰度变化值  $E(\mathbf{n})$  为  $\lambda_{\max}$

$$E(u, v) \cong [u, v] M \begin{bmatrix} u \\ v \end{bmatrix}$$

$$M\mathbf{x}_{\max} = \lambda_{\max}\mathbf{x}_{\max}; M\mathbf{x}_{\min} = \lambda_{\min}\mathbf{x}_{\min}; \mathbf{x}_{\max}^T\mathbf{x}_{\min}=0$$

$$\text{定义 } \mathbf{n} = [u, v]^T = a\mathbf{x}_{\max} + b\mathbf{x}_{\min}, \text{ with } a^2 + b^2 = 1$$

$$\text{于是, } E(\mathbf{n}) = \mathbf{n}^T M \mathbf{n}$$

$$= (a\mathbf{x}_{\max} + b\mathbf{x}_{\min})^T M (a\mathbf{x}_{\max} + b\mathbf{x}_{\min})$$

$$= (a\mathbf{x}_{\max} + b\mathbf{x}_{\min})^T (a\lambda_{\max}\mathbf{x}_{\max} + b\lambda_{\min}\mathbf{x}_{\min})$$

$$= a^2\lambda_{\max} + b^2\lambda_{\min}$$

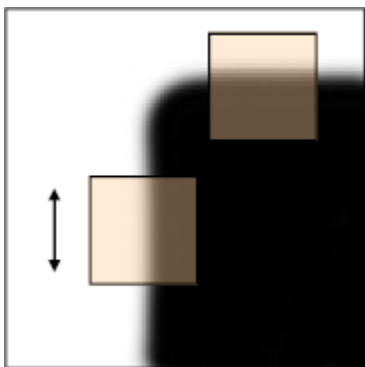
$$= a^2\lambda_{\max} + (1 - a^2)\lambda_{\min}$$

$$\text{因此, } E_{\max} = \lambda_{\max}, \text{ with } a = 1 \text{ and } \mathbf{n} = [u, v]^T = \mathbf{x}_{\max};$$

$$E_{\min} = \lambda_{\min}, \text{ with } a = 0 \text{ and } \mathbf{n} = [u, v]^T = \mathbf{x}_{\min}.$$

# 角点响应函数

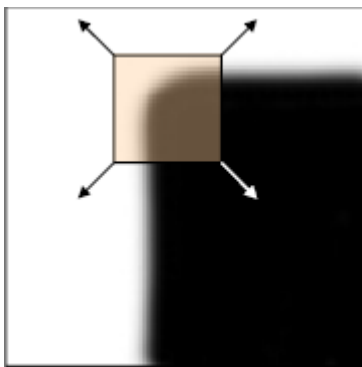
□ 定性分析矩阵对称矩阵M的特征值



**边缘:**

$$\lambda_1 \gg \lambda_2$$

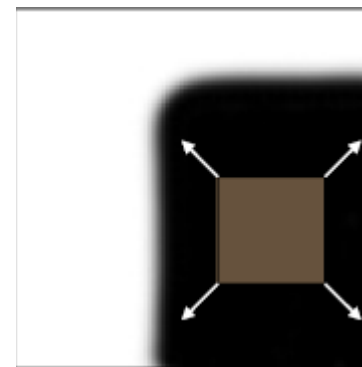
$$\lambda_2 \gg \lambda_1$$



**角点:**

$\lambda_1$  和  $\lambda_2$  均较大,

$$\lambda_1 \sim \lambda_2;$$

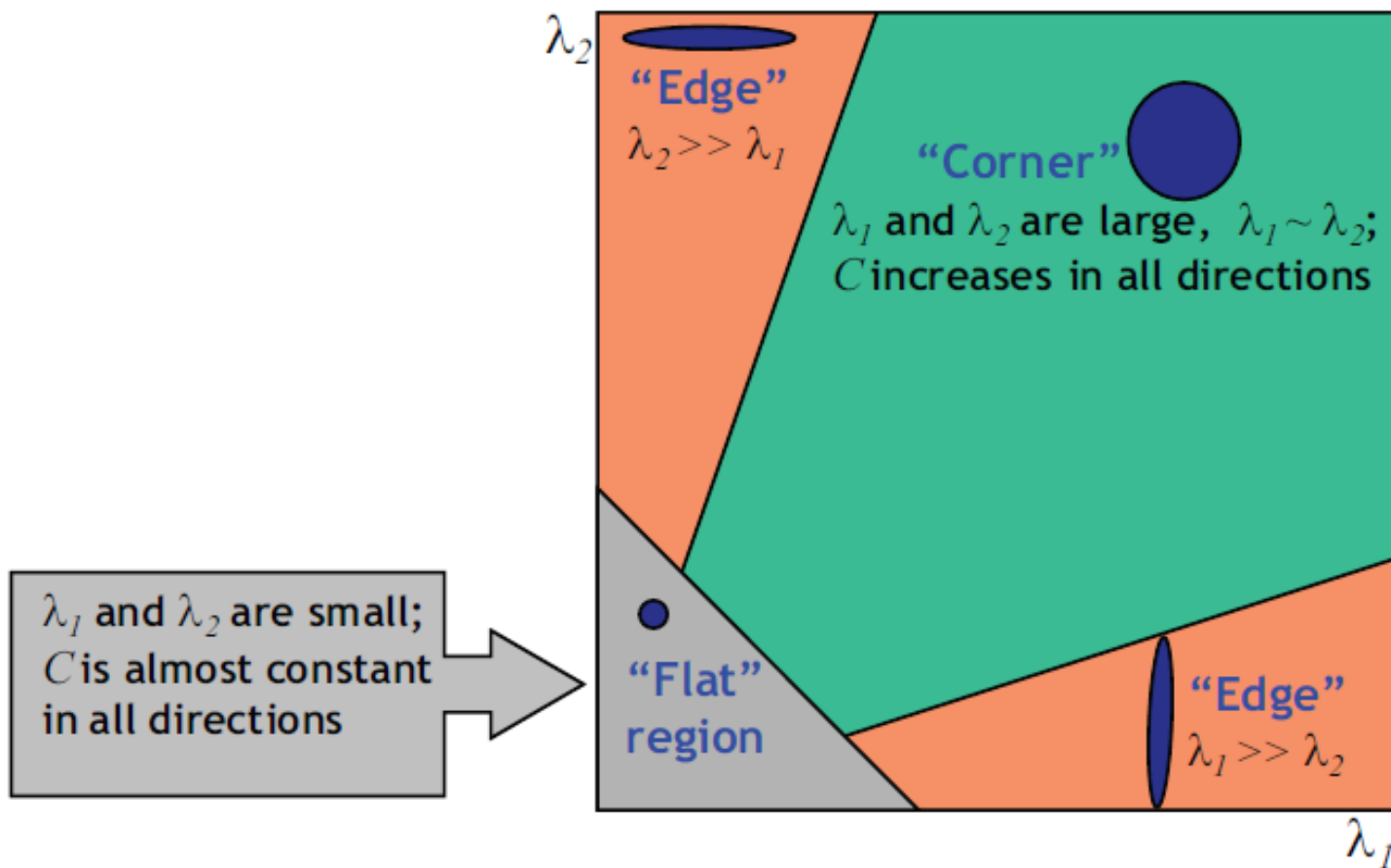


**平滑区域:**

$\lambda_1$  和  $\lambda_2$  均较小;

# 角点响应函数

$$\mathbf{M} = \begin{bmatrix} \sum_{(x,y) \in W} I_x^2 & \sum_{(x,y) \in W} I_x I_y \\ \sum_{(x,y) \in W} I_y I_x & \sum_{(x,y) \in W} I_y^2 \end{bmatrix} = \mathbf{R}^{-1} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \mathbf{R}$$





# 角点响应函数

- 对每个图像窗，计算 M 矩阵，基于其特征值，定义角点响应函数 R:

$$R(x, y) = \lambda_1 \lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$$

$$\det(\mathbf{C}) = \lambda_1 \lambda_2$$

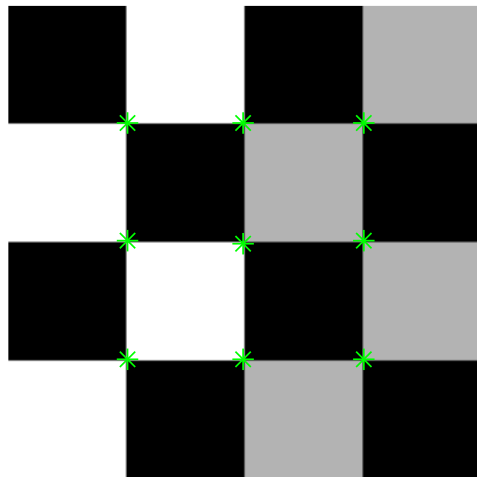
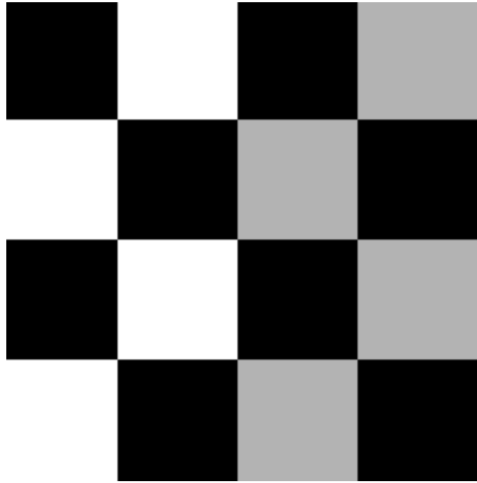
$$\text{trace}(\mathbf{C}) = \lambda_1 + \lambda_2$$

- 找到角点响应较大的值所对应的图像位置点:

$$R > \text{threshold}$$

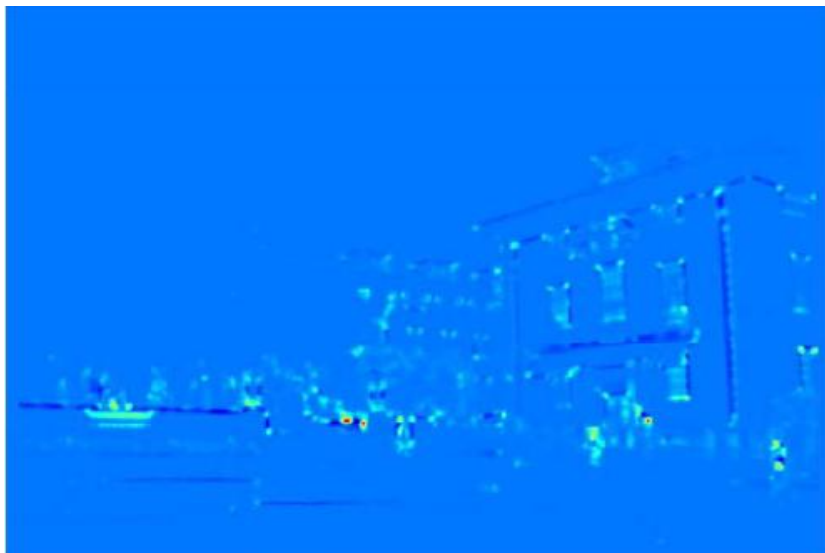
- 取 R 的局部极大值点，例如 进行非最大抑制

# Harris角点检测示例



# Harris角点检测示例角点响应图

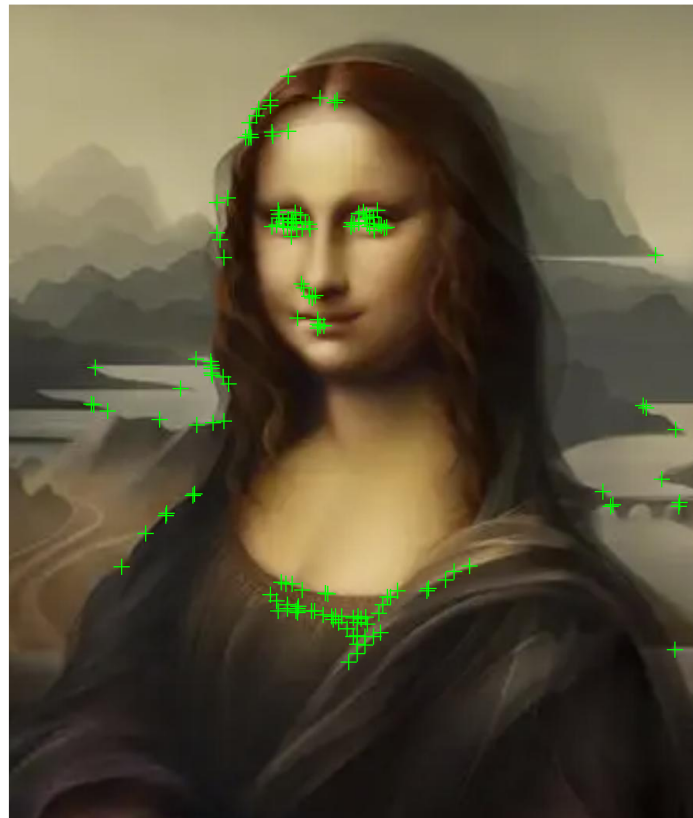
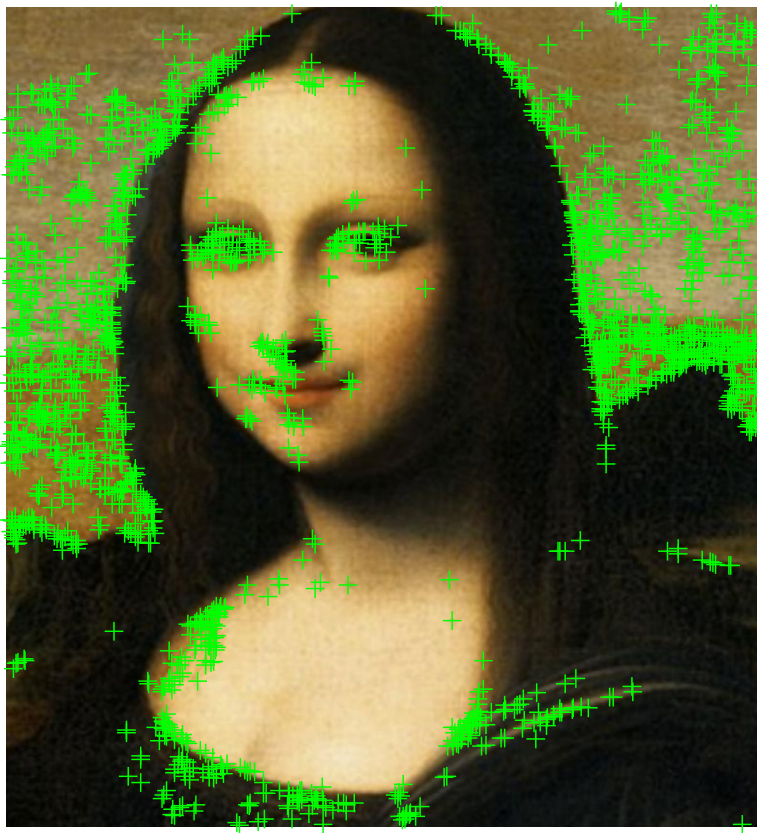
角点响应图  $R$



对  $R$  进行门限处理, 取极大值



# 对两张图像独立进行检测处理



# Harris 角点检测子的性质

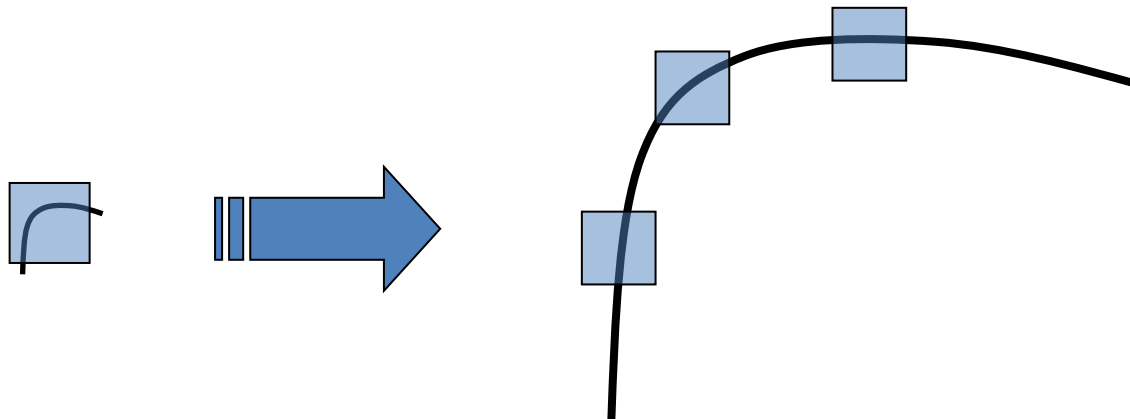
## □ 旋转不变?

- Yes: 因为旋转不会改变特征值, 只会影响特征向量的方向

$$M = X \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} X^T$$

## □ 尺度不变?

- No: 当把一个角点区域无限放大, 角点将变为边缘
- 如何得到角点的尺度?



# 校正窗口尺度函数

□ Harris 检测算子检测到的特征点 取决于所选的窗的大小



Window scale = 10



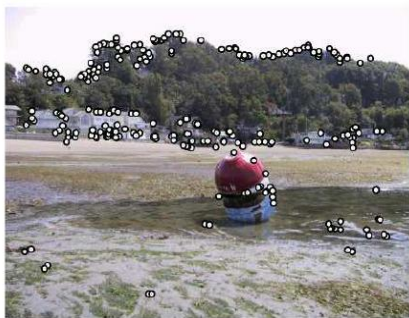
Window scale = 15



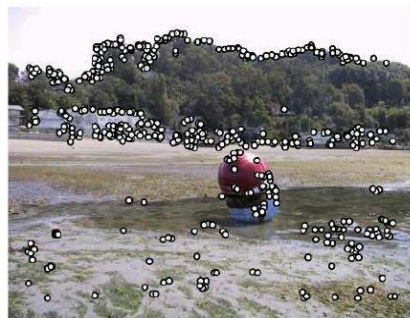
Window scale = 30

□ 固定窗口尺寸，施加自适应非极大抑制

- Adaptive non-maximal suppression (ANMS): 局部极大，同时响应值比邻域（半径 $r$ 的圆形区域）中其他像素响应最大值大10%



(a) Strongest 250



(b) Strongest 500



(c) ANMS 250,  $r = 24$



(d) ANMS 500,  $r = 16$



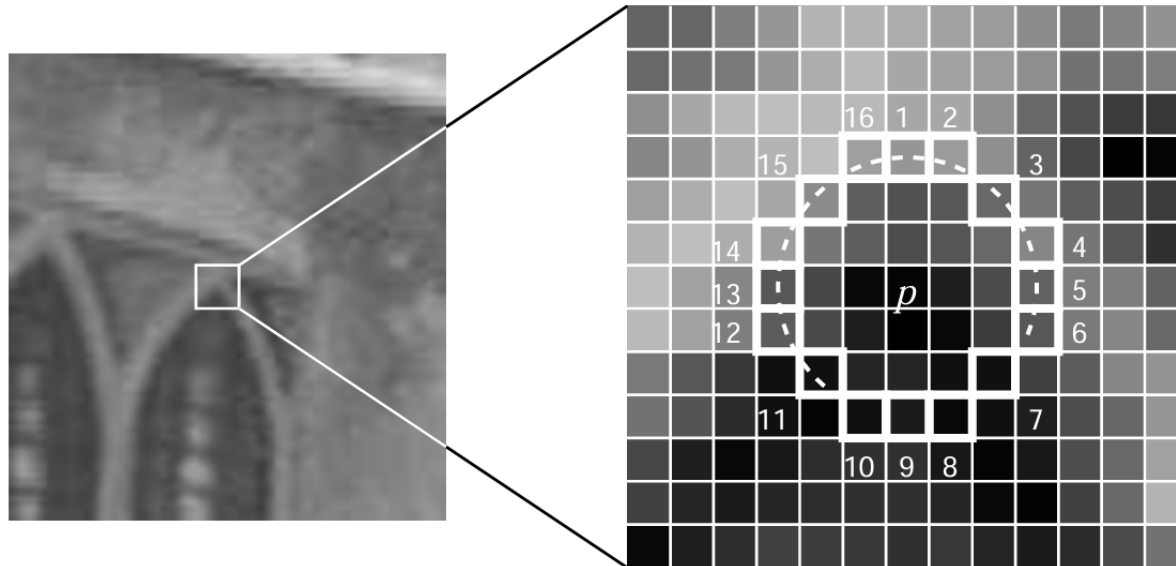
# 图像表达-关键点检测

- 角点检测 (Corner Detection)
  - Harris检测子
  - **FAST**检测子
- 块检测 (Blob Detection)
  - Laplacian-of-Gaussian (LoG) 检测子
  - Difference-of-Gaussian (DoG)检测子
  - SURF检测子
  - MSER检测子

# FAST: 快速角点检测子

## □ FAST: Features from Accelerated Segment Test

- 通过分析像素点与其邻域内像素点的灰度关系，来判断其是否为角点
- 例如，如下图，对于像素 $p$ ，在其邻域圆形窗口上选16个像素
  - ✓ 如果有 $n$ 个连续像素点的灰度显著都大于或都小于中心像素，则 $p$ 为角点
  - ✓ 为了提高检测速度，可以先检测圆上的1、9、5、13四个像素点
  - ✓ 若这四个点中至少三个点的灰度与中心像素点明显不同，则 $p$ 可能为角点，然后再对所有16个像素点进行检测以确认
- 为进一步提升性能，采用机器学习方法提升速度和通用性



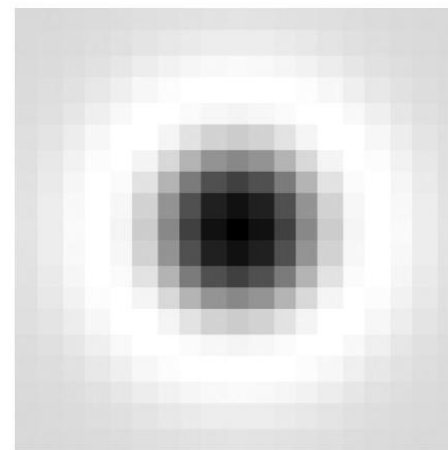
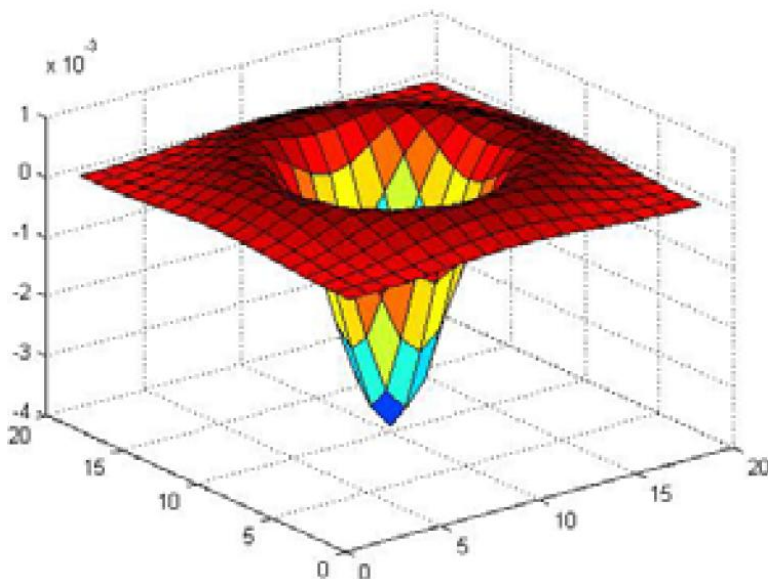


# 图像表达-关键点检测

- 角点检测 (Corner Detection)
  - Harris检测子
  - FAST检测子
- 块检测 (Blob Detection)
  - **Laplacian-of-Gaussian (LoG) 检测子**
  - Difference-of-Gaussian (DoG)检测子
  - SURF检测子
  - MSER检测子

# Laplacian-of-Gaussian (LoG)

- 2D的圆周对称操作子，用于块(blob)检测
  - LoG: “blob” detector

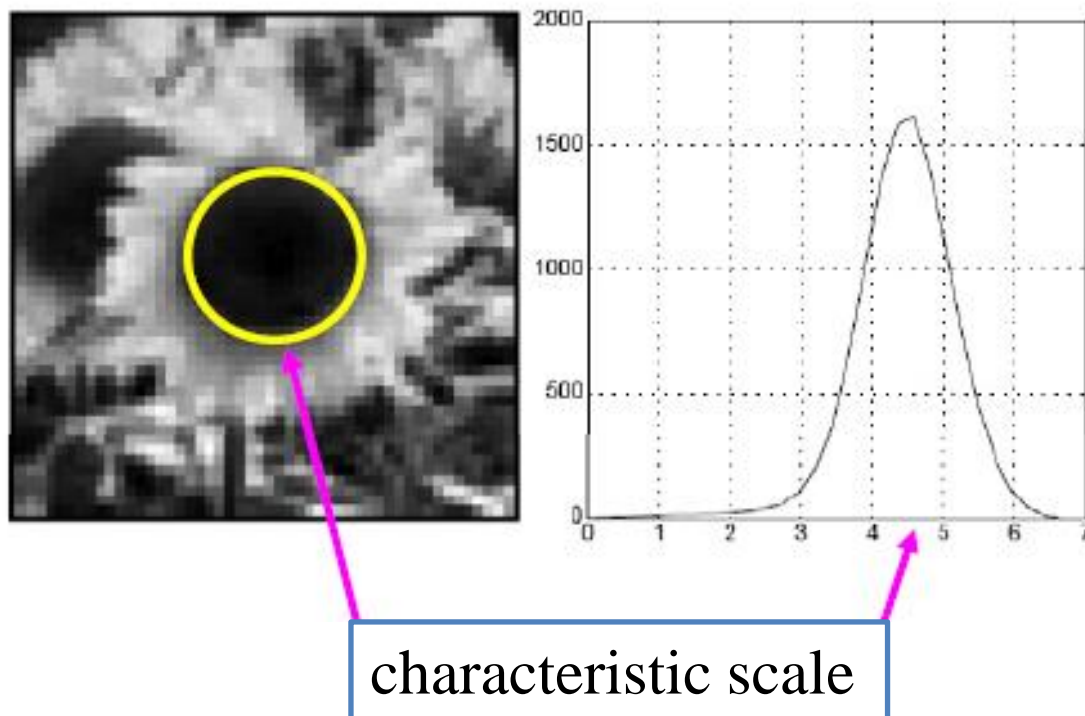


$$LoG = \nabla^2 G_\sigma(x, y) = \frac{\partial^2}{\partial x^2} G_\sigma(x, y) + \frac{\partial^2}{\partial y^2} G_\sigma(x, y)$$

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

# 对于图像中的一个给定点:

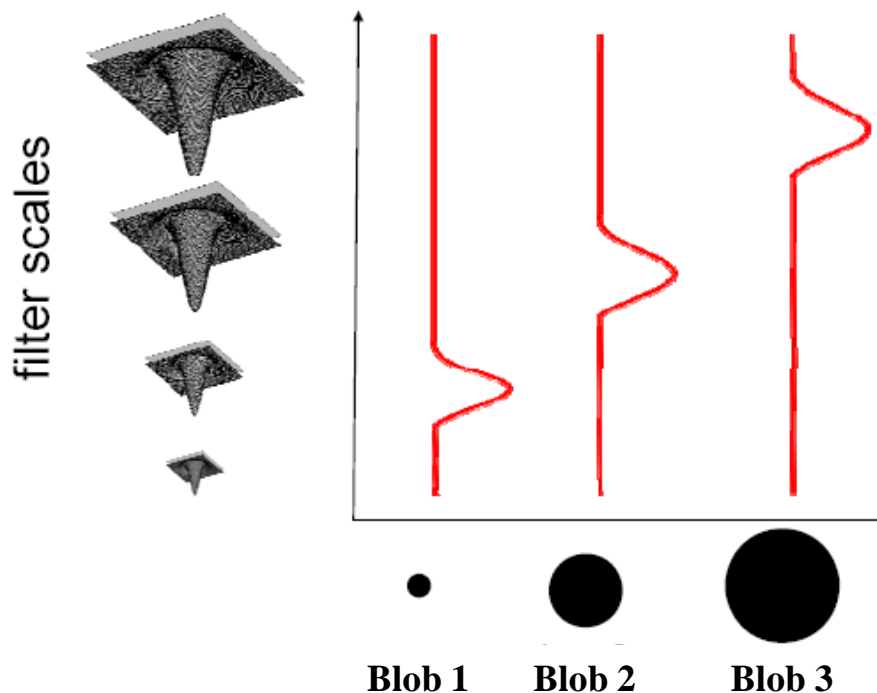
- 特征尺度 (characteristic scale)
  - Laplacian 响应的极值所对应的尺度



# Laplacian-of-Gaussian (LoG)

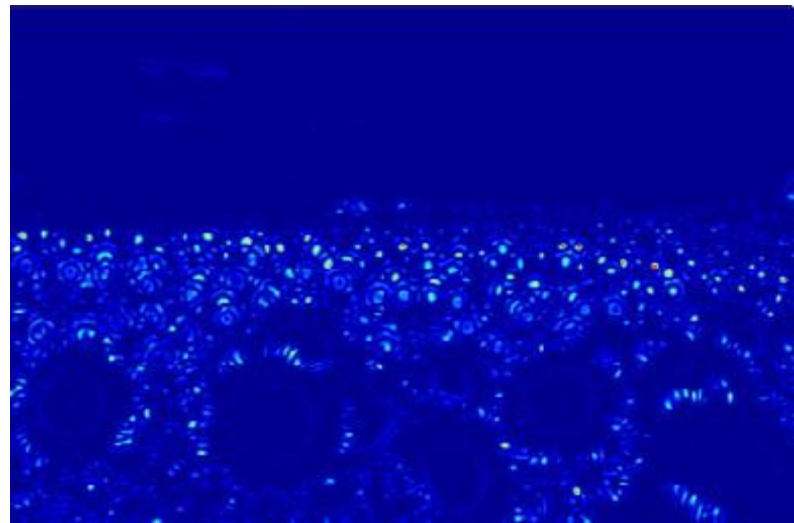
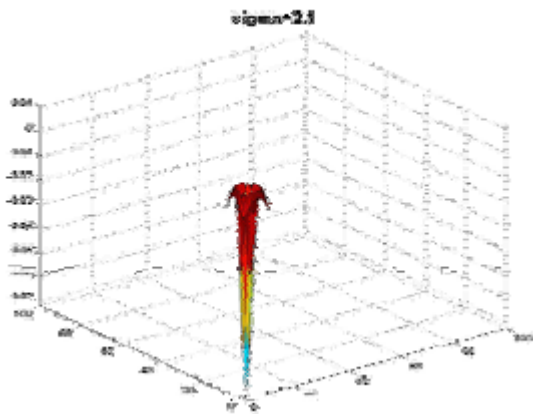
## □ Laplacian-of-Gaussian = “blob” detector

- 给定图像块 (patch)，遍历枚举不同尺寸LoG的滤波器，将每个滤波器与图像块相卷积
- 根据卷积响应的极大/极小值，确定适合当前图像块的LoG的滤波器，该滤波器的尺寸即定义了图像的尺度

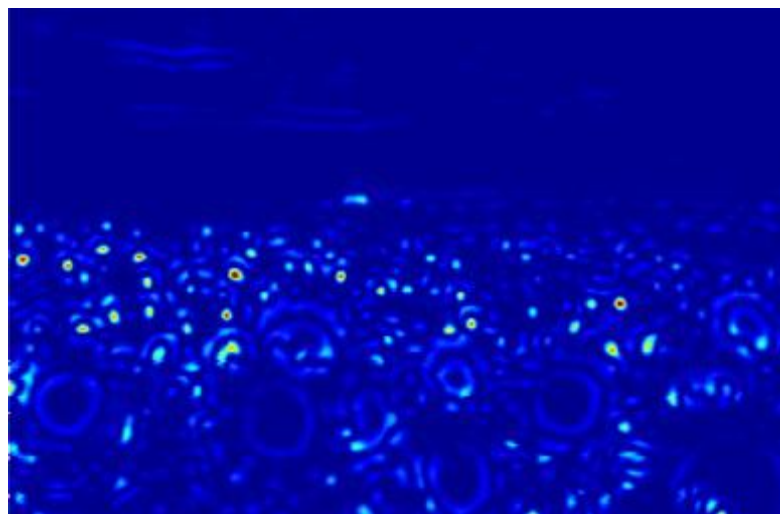
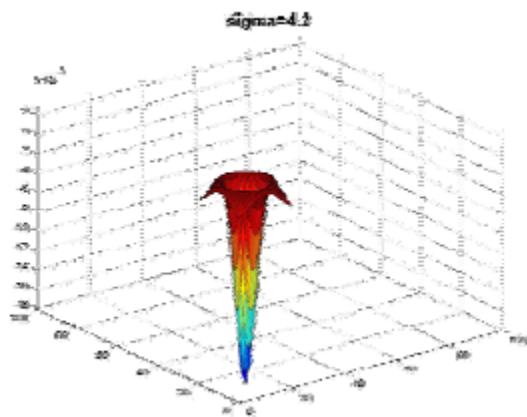


$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

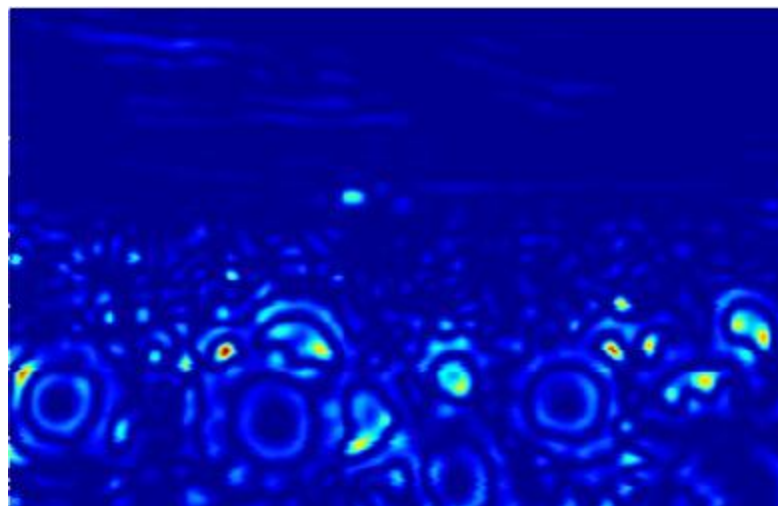
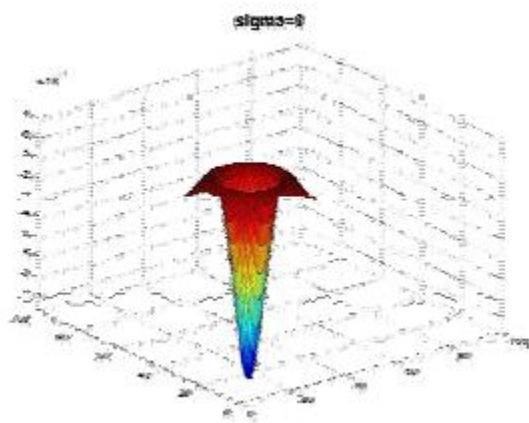
# LoG响应示意图-1



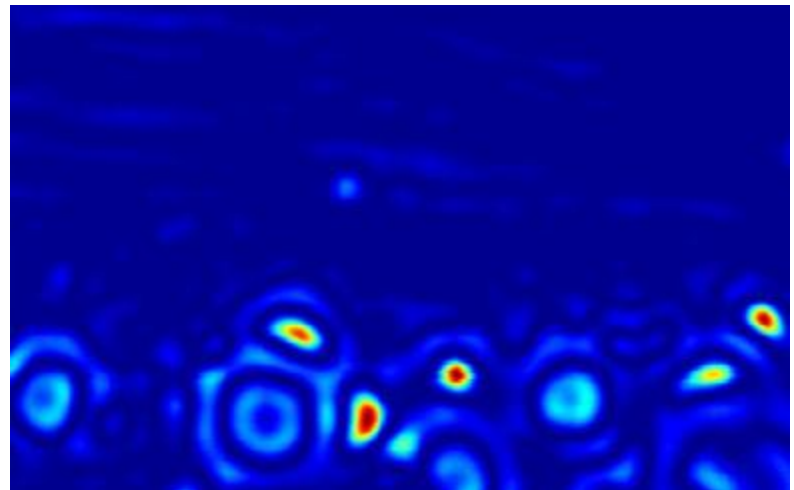
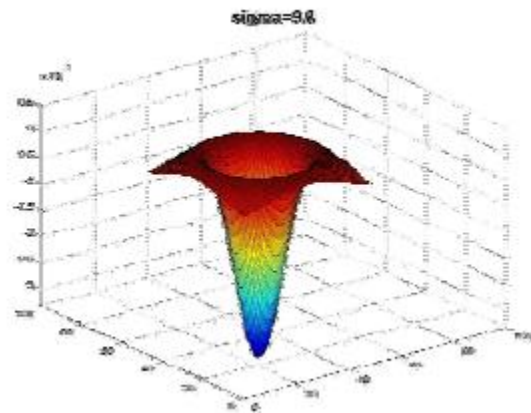
# LoG响应示意图-2



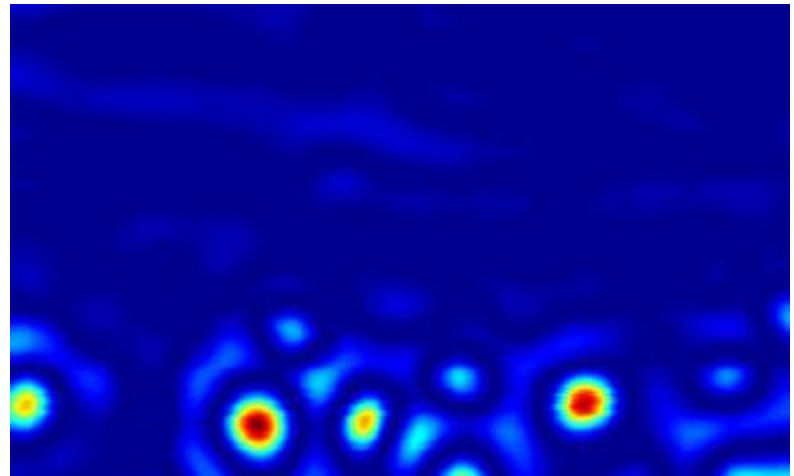
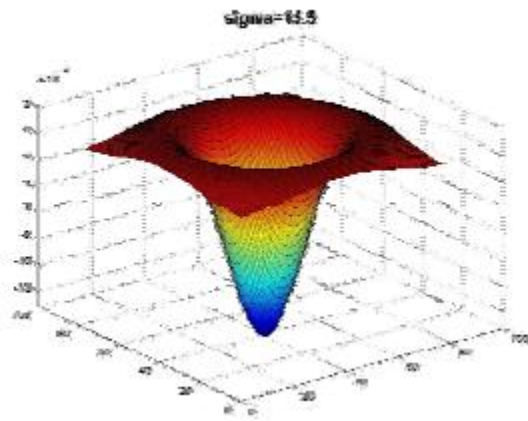
# LoG响应示意图-3



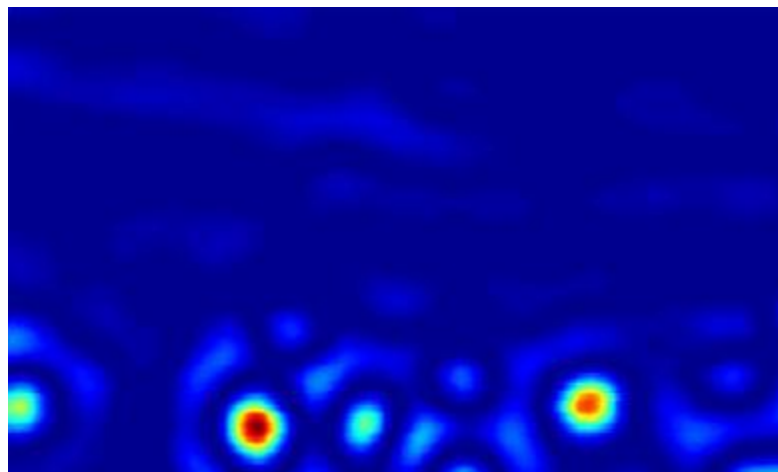
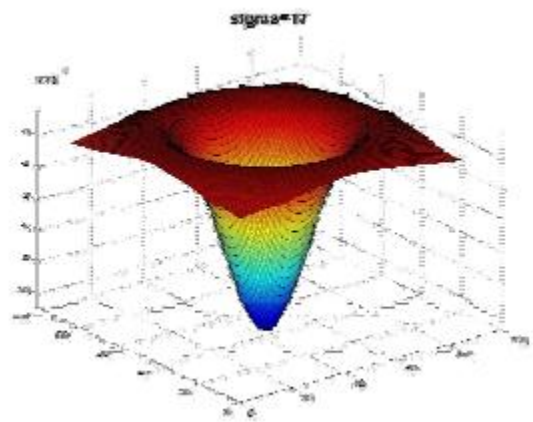
# LoG响应示意图-4



# LoG响应示意图-5

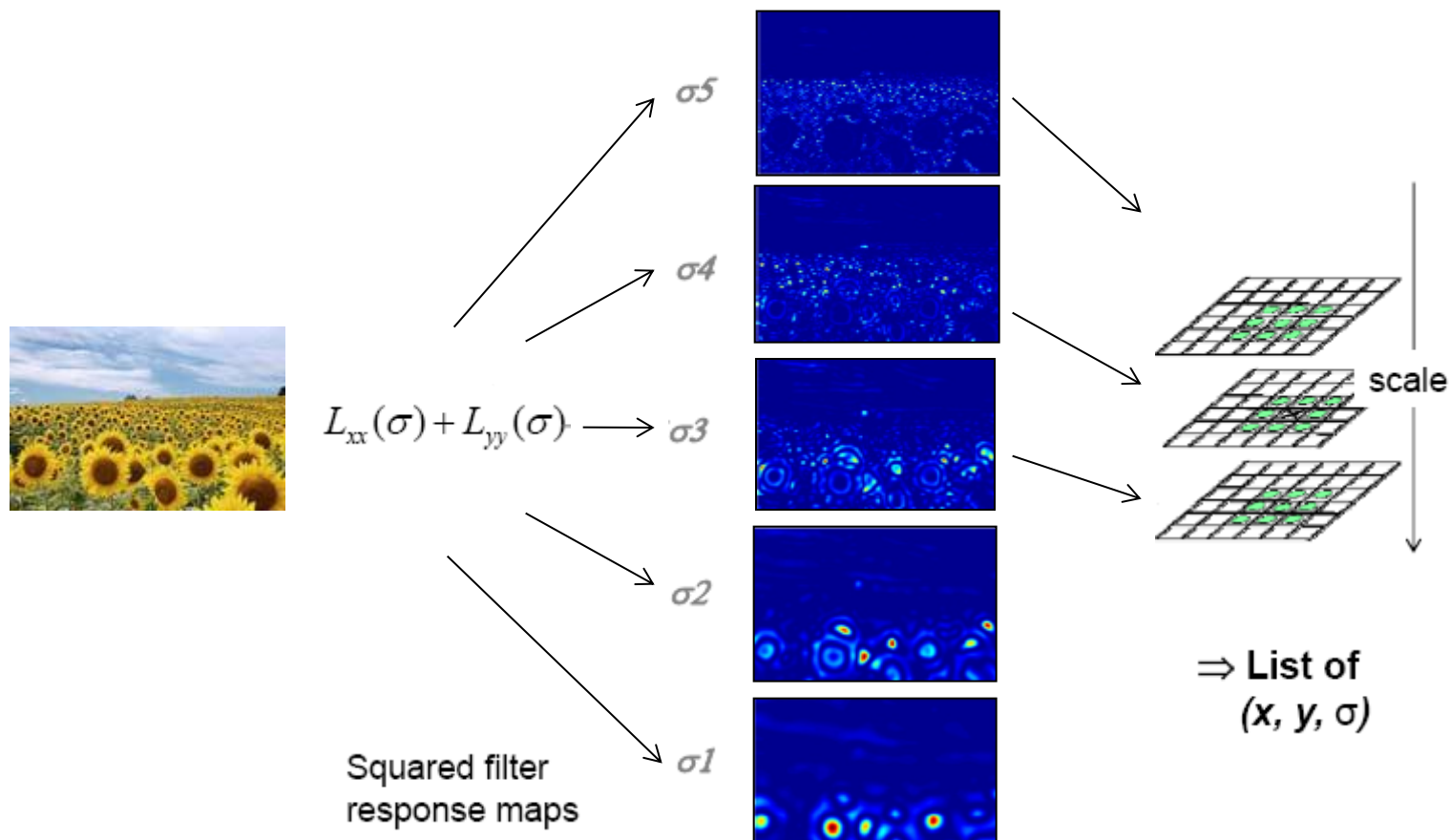


# LoG响应示意图-6



# 尺度空间的 blob 检测

- 根据卷积响应的极大/极小值，确定适合当前像素位置的 LoG 的滤波器，该滤波器的尺寸即定义了当前像素的尺度



# 尺度空间的 blob 检测: 示例

- 检测出来的区域不一定是完整的blob或独立的区域





# 图像表达-关键点检测

## □ 角点检测 (Corner Detection)

- Harris检测子
- FAST检测子

## □ 块检测 (Blob Detection)

- Laplacian-of-Gaussian (LoG) 检测子
- **Difference-of-Gaussian (DoG)检测子**
- SURF检测子
- MSER检测子

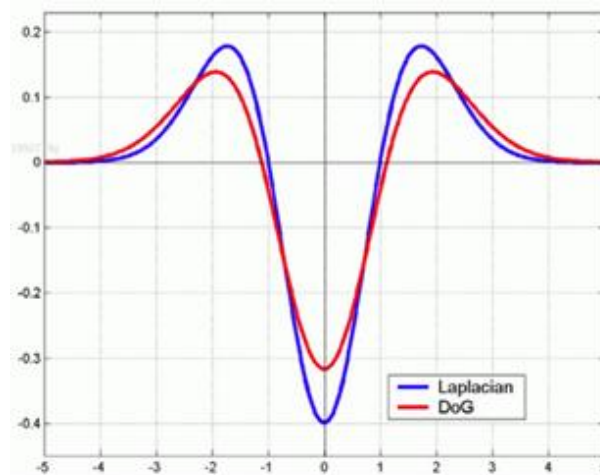
# 从LoG到DoG: SIFT检测子

□ 采用 Difference of Gaussians (DoG) 去近似 Laplacian of Gaussians (LoG)

- 差分阶次更低 (二阶 → 一阶)
- 高效的实现形式

LoG:  $L = \sigma^2 (G_{xx}(x, y, \sigma) + G_{yy}(x, y, \sigma))$

DoG:  $DoG = G(x, y, k\sigma) - G(x, y, \sigma)$

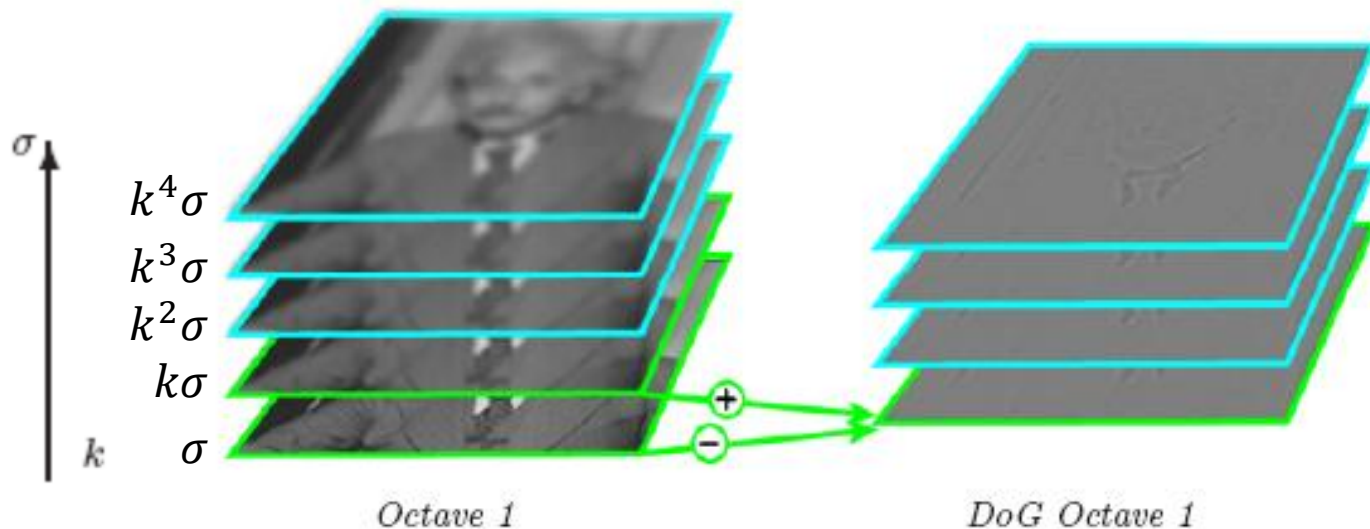


# DoG 图像金字塔-1

## □ 首先构建高斯金字塔

- 用尺度为 $\sigma$ 的高斯核对图像做卷积，得到不同尺度的图像
- 每层octave有 $s + 3$ 副滤波结果图像， $k^s = 2$ ，即 $k = 2^{\frac{1}{s}}$
- 相邻高斯图作差，得到高斯差分图，每层octave有 $s + 2$ 张图
  - ✓ 节省计算开销

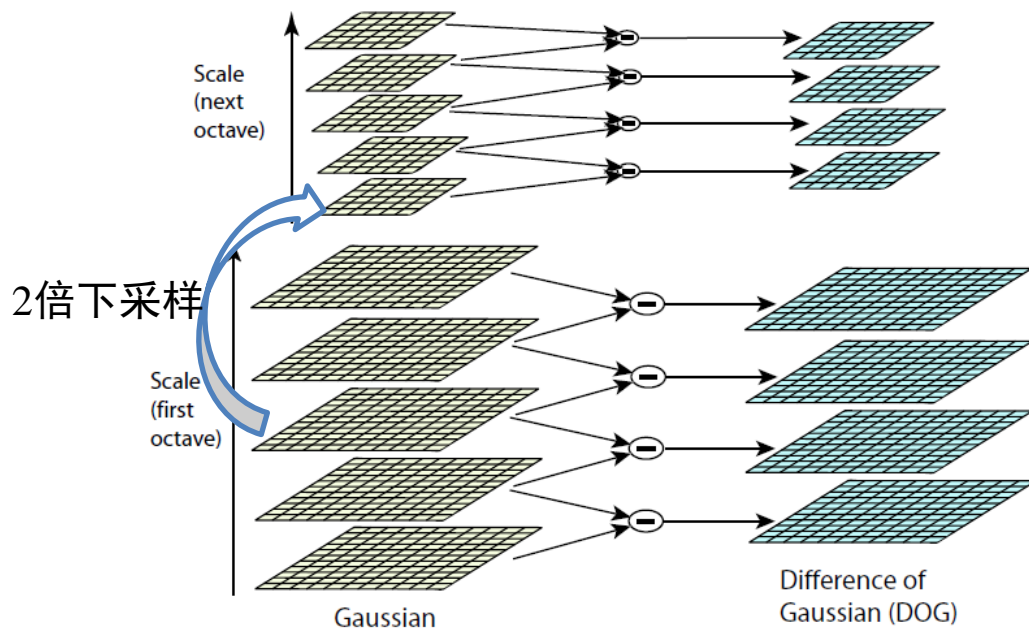
$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned}$$



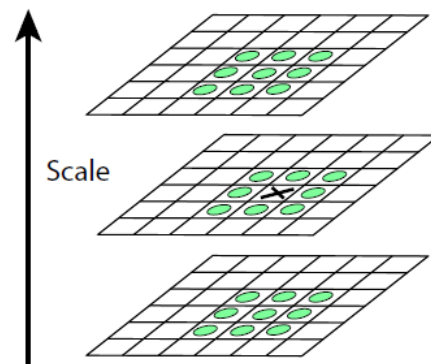
# DoG 图像金字塔-2

## □ 关键点检测

- 相邻两个octave的尺度差为2
- 每组octave的 $s + 2$ 副DoG图表示  $(x, y, \text{scale})$  三维尺度空间
- 在 $3 \times 3 \times 3$ 的26邻域寻找局部极大、极小值点
- 对于每个octave, 局部极值点仅落在 $s$ 副DoG图像上



构建多级octave, 相邻两级下采样率为2



在 $3 \times 3 \times 3$ 的26邻域, 检测极值点



# DoG 图像金字塔-3

## □ 小结

- 将图象中的关键区域的尺度估计问题转换为DoG滤波器匹配问题，将最匹配的DoG滤波器的标准差作为关键区域的尺度估计值
- 将图象的尺度空间划分为一系列子区间： $(\sigma, 2\sigma]$ ,  $(2\sigma, 4\sigma]$ ,  $(4\sigma, 8\sigma]$ ,  $\dots$ ,  $(2^m\sigma, 2^{m+1}\sigma]$ ,  $\dots$ 。每个尺度子区间对应一个octave
- 由于相邻两个octave对图象进行了2倍（行、列）下采样，所以每个octave中的都等价于在不同图象分辨率下进行尺度范围为 $[\sigma, 2\sigma)$ 的关键区域尺度估计
- 在第 $m$ 个octave上的尺度估计结果，通过乘以 $2^m$ ，即可得到在原始图象分辨率下的关键区域尺度估计值



# 图像表达-关键点检测

- 角点检测 (Corner Detection)
  - Harris检测子
  - FAST检测子
- 块检测 (Blob Detection)
  - Laplacian-of-Gaussian (LoG) 检测子
  - Difference-of-Gaussian (DoG)检测子
  - **SURF检测子**
  - MSER检测子

# SURF检测子

## □ SURF (Speeded-Up Robust Features)

- 采用Hessian矩阵对图像滤波，基于行列式值选择关键点位置和尺度

$$\mathcal{H}(\mathbf{x}, \sigma) = \begin{bmatrix} L_{xx}(\mathbf{x}, \sigma) & L_{xy}(\mathbf{x}, \sigma) \\ L_{xy}(\mathbf{x}, \sigma) & L_{yy}(\mathbf{x}, \sigma) \end{bmatrix}$$

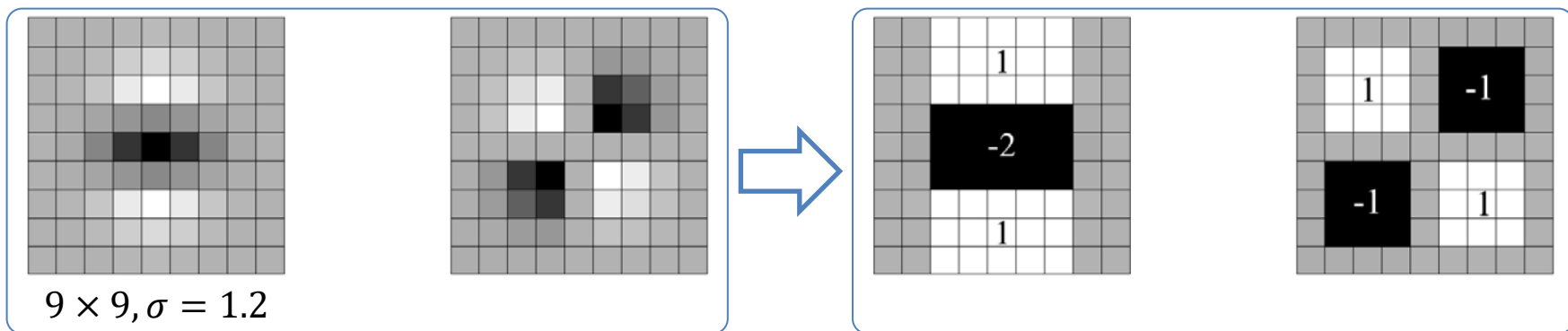
✓  $L_{xx}(\mathbf{x}, \sigma)$ 表示对图像 $I$ 在像素 $\mathbf{x}$ 处的高斯二阶微分 $\frac{\partial^2}{\partial x^2} g(\sigma)$ 的卷积

✓ 高斯函数是尺度空间分析的重要工具

- 高斯二阶微分可以用**方框滤波器** (box filter) 近似 ( $L_{xx} \rightarrow D_{xx}$ )

✓  $Det(\mathcal{H}_{approx}) = D_{xx}D_{yy} - (0.9D_{xy})^2$

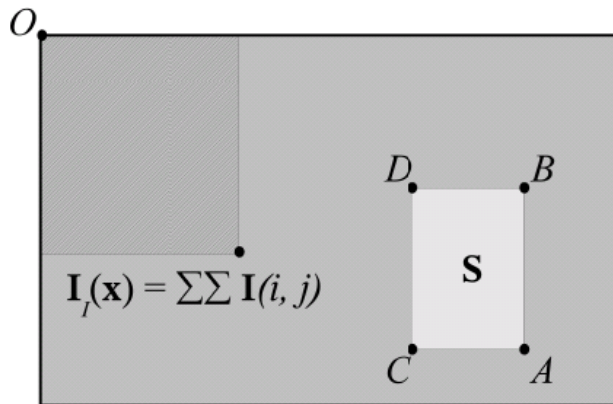
✓ 利用积分图可以显著提升计算效率



# SURF检测子

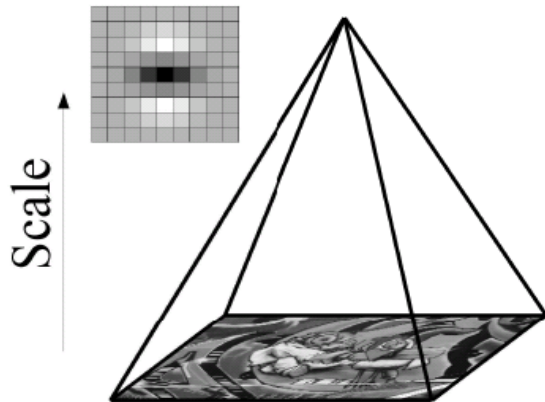
## □ 图像积分图

- 任意像素区域的灰度可通过区域四个顶点的积分图的值计算得到

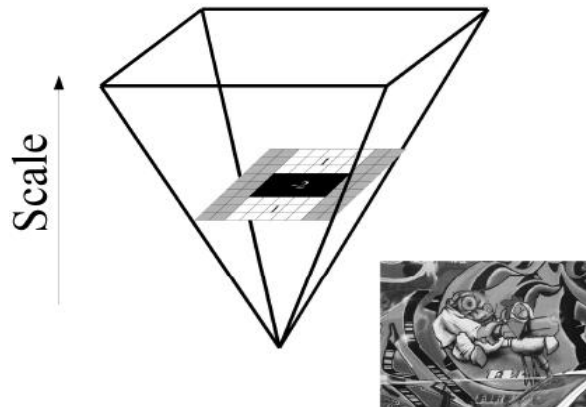


$$S = A - B - C + D$$

## □ 利用积分图计算优势，构建滤波器金字塔做尺度空间分析

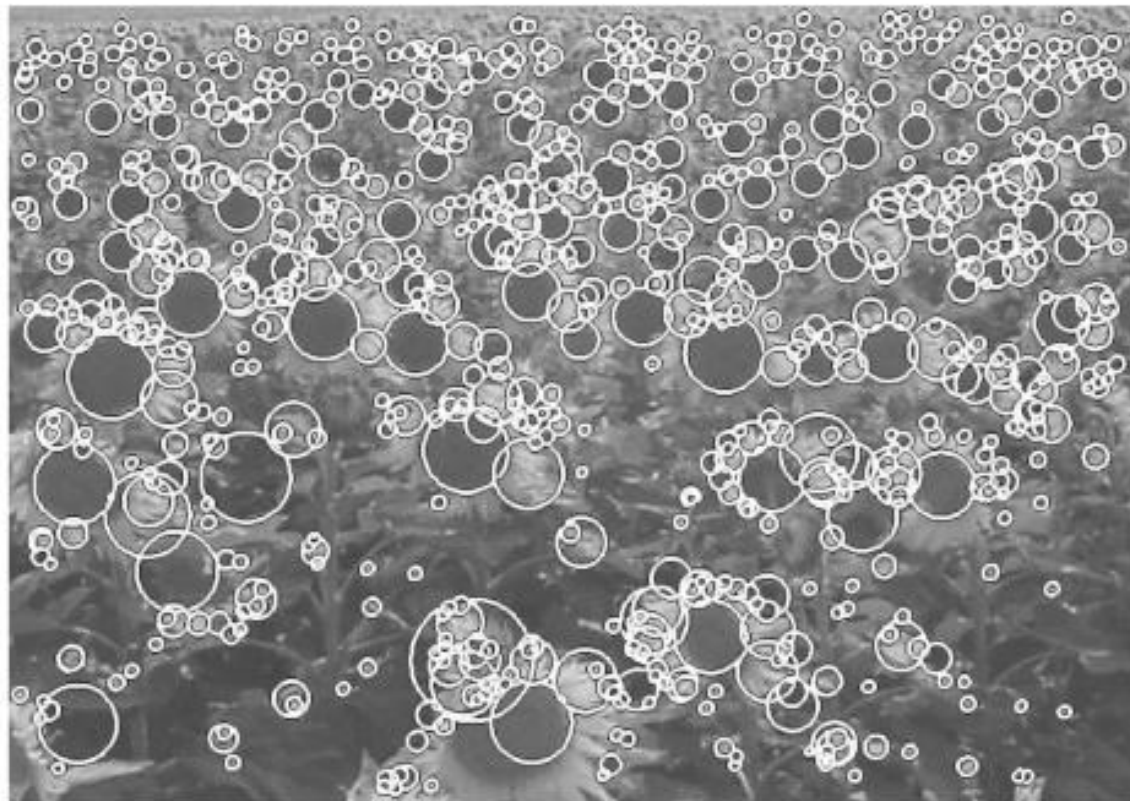
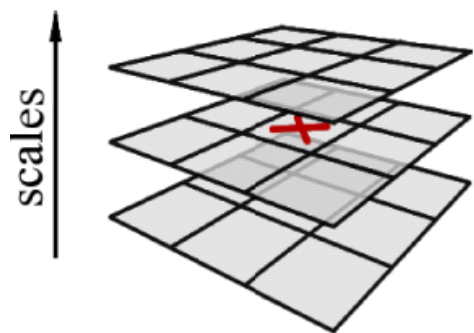


vs.



# SURF检测子

- 基于近似Hessian矩阵的行列式值，在尺度空间的 $3 \times 3 \times 3$ 邻域中计算局部极大值，并做非极大抑制（NMS）





# 图像表达-关键点检测

- 角点检测 (Corner Detection)
  - Harris检测子
  - FAST检测子
- 块检测 (Blob Detection)
  - Laplacian-of-Gaussian (LoG) 检测子
  - Difference-of-Gaussian (DoG)检测子
  - SURF检测子
  - **MSER检测子**

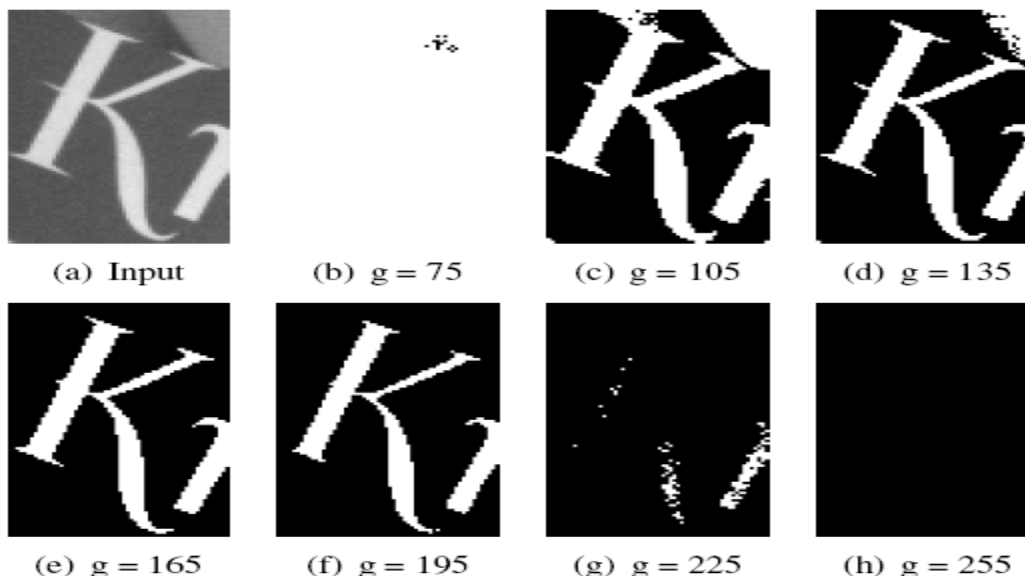
# 最大稳定极值区域检测子: MSER

## □ MSER: Maximally Stable Extremal Region

- 在一个比较大的阈值范围内, 可被独立分割出来的、面积稳定的图像区域
- 区域内部和区域边界上的像素的灰度有显著差异

$$\forall p \in R_i, \forall q \in \text{boundary}(R_i) \rightarrow I(p) - I(q) > \tau$$

- 实现时, 可用一系列渐变的阈值对图像进行分割, 比较分割结果

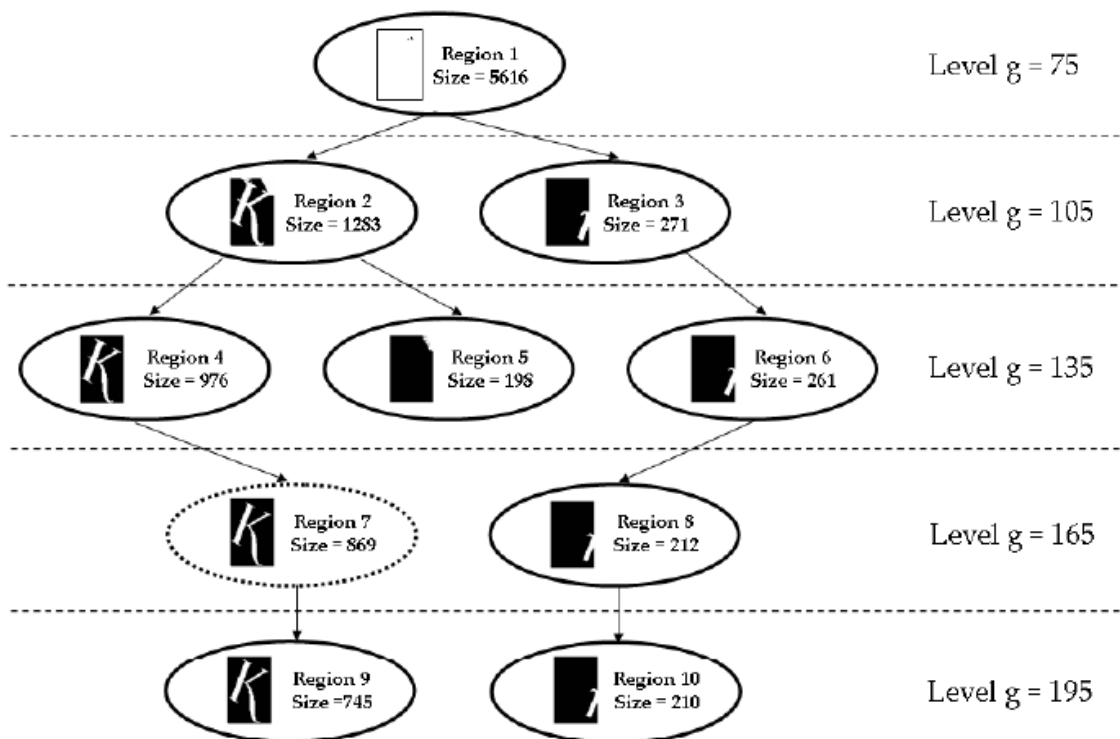


• Matas J, et al. Robust wide-baseline stereo from maximally stable extremal regions[J]. Image and Vision Computing, 2004.

# 最大稳定极值区域检测子: MSER

## □ 对比不同阈值下的分割结果，构建分割树

- 该树结构中，不同层对应不同的分割阈值
- 对各个叶节点，往上回溯到其第一个分叉节点（子节点数量大于2的节点），统计回溯的层级数；若层级数大于指定阈值，且区域面积变化较小，则该区域为一个MSER



# 最大稳定极值区域检测子: MSER

## □ MSER的特点

- 对较大仿射畸变有较好的检测鲁棒性，
  - ✓ 在可重复性 (repeatability) 指标上，优于其他局部特征检测子
- 存在的问题：一副图象中检测得到的MSER区域数量有限
  - ✓ 例如，对于一副 $640 \times 480$ 的图像，MSER数量一般 $< 100$
  - ✓ 可以支持的图像之间的MSER匹配数量有限

