

# 第3章 图像分析— 3.2点描述

信控学院 蔡利梅

**图像描述：**将分割后区域的区域、边界的属性和相互关系用更为简单明确的文字、数值、符号或图来描述或说明。  
**保留原图像或图像区域重要信息，减少数据量**

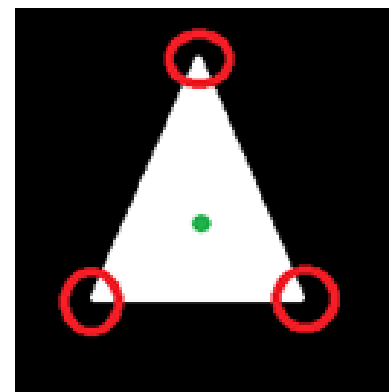
**图像的描述子：**按一定的概念或公式从图像中产生文字、数值、符号或图，也称为**图像的特征**；产生它们的过程称为**图像特征提取**。

**应具有的特点：**唯一性、完整性、几何变换不变性、敏感性、抽象性。

- 点描述
- 区域描述
- 边界描述
- 纹理描述

### 3.2.1 基本概念

**特征点：**典型的特征标志之一，一般含有显著的结构性信息，如线条交叉点、边界封闭区域的重心，或者曲面的高点等；也可以没有实际的直观视觉意义，但在某种角度、某个尺度上含有丰富的易于匹配的信息。

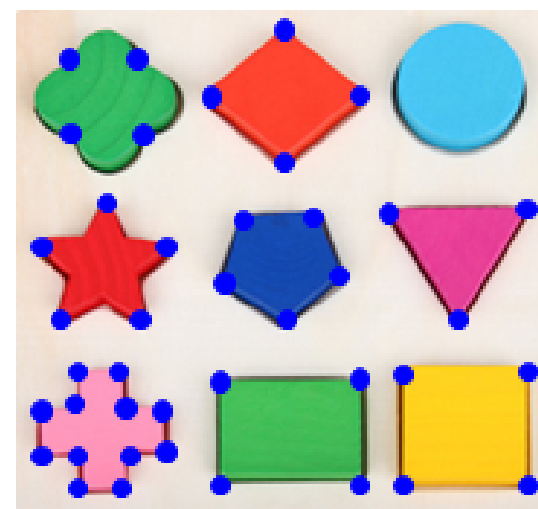


特征点

**应用：**在影像匹配、图像拼接、运动估计以及形状描述等方面都具有重要作用。

**角点：**特征点中最主要的一类，由景物曲率较大地方的两条或多条边缘的交点形成，如线段的末端、轮廓的拐角。

角点特征与直线、圆、边缘等其它特征相比，具有**提取过程简单**、**结果稳定**、**提取算法适应性强**的特点，成为特征匹配算法的首选。



## 3.2.2 角点检测

### ■ Moravec角点检测

以图像某个像素点为中心，计算固定窗口内四个主要方向上（水平、垂直、对角线、反对角线）**相邻像素灰度差的平方和，选取最小值**作为像素点的响应函数CRF（Corner Response Function）；若某点的CRF值大于某个阈值并为局部极大值时，则该像素点即为角点。

## ■ Moravec角点检测

固定窗口在平坦区域时，灰度比较均匀，4个方向的灰度变化值都很小，CRF也小

$$\nabla_h = 0$$

$$\nabla_v = 0$$

$$\nabla_{d1} = 0$$

$$\nabla_{d2} = 0$$

$$CRF=0$$

1	1	1	1	1	1
1	1	6	1	1	1
1	6	1	6	1	1
6	1	1	1	6	1
1	1	1	1	1	6
1	1	1	1	1	1

在边缘处，沿边缘方向的灰度变化值很小，沿垂直边缘方向的灰度变化值比较大，CRF小

1	1	1	1	1	1
1	1	6	1	1	1
1	6	1	6	1	1
6	1	1	1	6	1
1	1	1	1	1	6
1	1	1	1	1	1

$$\nabla_h = 50$$

$$\nabla_v = 50$$

$$\nabla_{d1} = 50$$

$$\nabla_{d2} = 0$$

$$CRF=0$$

## ■ Moravec角点检测

当窗口在角点或独立点上的时候，沿各个方向的灰度变化值都比较大，CRF也大

$$\nabla_h = 50$$

$$\nabla_v = 50$$

$$\nabla_{d1} = 25$$

$$\nabla_{d2} = 25$$

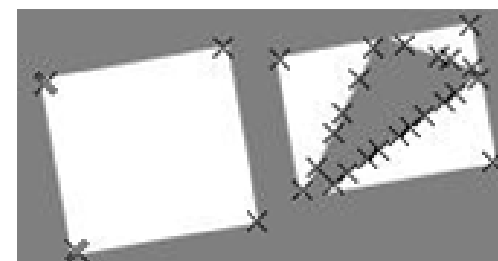
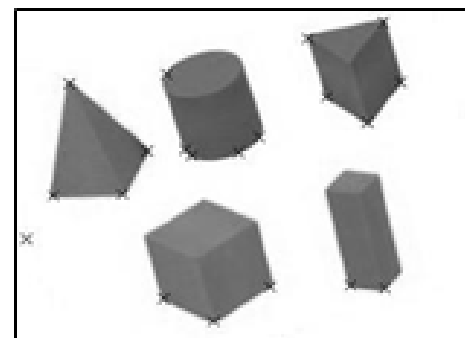
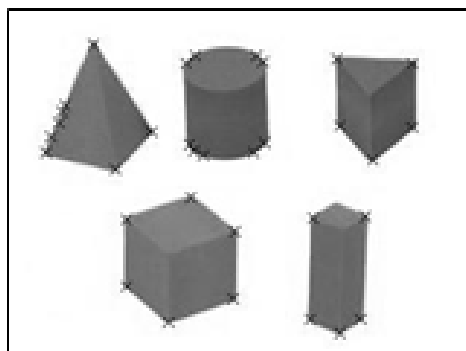
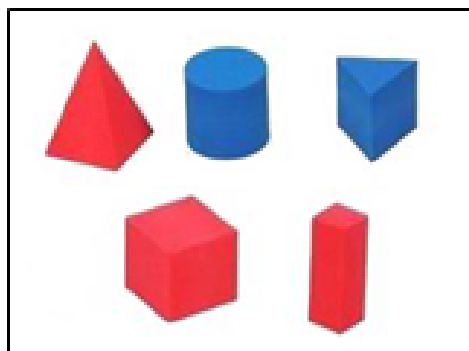
$$CRF = 25$$

1	1	1	1	1	1
1	1	6	1	1	1
1	6	1	6	1	1
6	1	1	1	6	1
1	1	1	1	1	6
1	1	1	1	1	1

若某窗口内各个方向变化的最小值大于某个阈值，说明各方向的变化都比较大，则该窗口所在即为角点所在。

1	1	1	1	1	1
1	1	6	1	1	1
1	6	1	6	1	1
6	1	1	1	6	1
1	1	1	1	1	6
1	1	1	1	1	1

## ■ Moravec角点检测



测试图

Moravec角点检测

旋转后检测

Moravec角点检测对边缘点比较敏感，检测结果受到阈值的极大影响，且不具有旋转不变性。



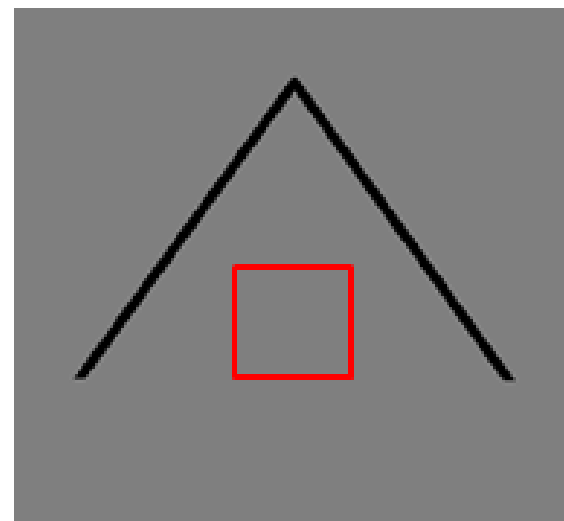
## ■ Harris角点检测

### □ 原理

局部自相关函数:  $E(\Delta x, \Delta y) = \sum_{x,y} w(x,y) [f(x+\Delta x, y+\Delta y) - f(x,y)]^2$

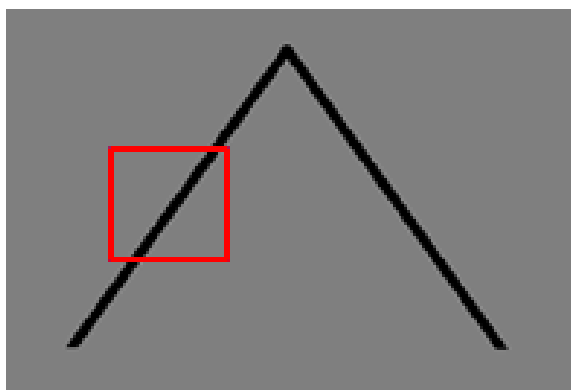
移动局部窗口，根据局部自相关函数的变化确定角点

**窗口在平坦区域：**沿任何方向进行小的平移，灰度变化很小，局部自相关函数很平坦

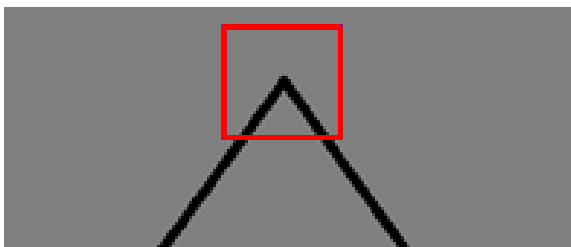


## ■ Harris角点检测

### □ 原理



**窗口位于边缘区域：**沿边缘方向小的平移，灰度变化很小；沿垂直边缘方向小的移动，灰度变化很大，局部自相关函数呈现山脊形状



**窗口位于角点区域：**窗口在各个方向上小的移动，灰度变化都很明显，局部自相关函数呈现尖峰状

角点检测即是寻找随着 $\Delta x$ 、 $\Delta y$ 的变化，局部自相关函数 $E(\Delta x, \Delta y)$ 的变化都比较大的像素点。

## ■ Harris角点检测

### □ 求解局部自相关函数

取 $f(x + \Delta x, y + \Delta y)$ 泰勒级数展开的一阶近似，自相关函数为：

$$E(\Delta x, \Delta y) = (\Delta x \quad \Delta y) \sum_{x,y} w(x,y) \begin{vmatrix} f_x f_x & f_x f_y \\ f_x f_y & f_y f_y \end{vmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}$$

$f_x, f_y$ : 图像水平和垂直方向的梯度

$$M = \sum_{x,y} w(x,y) \begin{vmatrix} f_x f_x & f_x f_y \\ f_x f_y & f_y f_y \end{vmatrix} = w * \begin{vmatrix} f_x f_x & f_x f_y \\ f_x f_y & f_y f_y \end{vmatrix} = \begin{vmatrix} A & C \\ C & B \end{vmatrix}$$

$$A = w(x,y) * f_x^2$$

$$B = w(x,y) * f_y^2$$

$$C = w(x,y) * f_x f_y$$

$$E(\Delta x, \Delta y) \approx A \Delta x^2 + 2C \Delta x \Delta y + B \Delta y^2$$

令 $E$ =常数，一个椭圆

## ■ Harris角点检测

### □ 求解局部自相关函数

椭圆的长短轴是与 $M$ 的特征值 $\lambda_1$ 、 $\lambda_2$ 相对应的量，通过判断 $\lambda_1$ 、 $\lambda_2$ 的情况可以区分出平坦区域、边缘和角点三种情况：

- ◆ 平坦区域， $f_x$ 、 $f_y$ 都较小，对应 $\lambda_1$ 、 $\lambda_2$ 都小，自相关函数 $E$ 在各个方向上取值都小；
- ◆ 边缘区域， $f_x$ 、 $f_y$ 只有一个较大，对应 $\lambda_1$ 、 $\lambda_2$ 一个较大，一个较小，自相关函数 $E$ 在某一方向上大，其他方向上小；
- ◆ 角点， $f_x$ 、 $f_y$ 都较大，对应 $\lambda_1$ 、 $\lambda_2$ 都大且近似相等，自相关函数 $E$ 在所有方向都增大。

## ■ Harris角点检测

### □ 求解局部自相关函数

为避免特征值直接求解，提高检测效率，设计角点响应函数：

$$R = \det M - k(\text{trace} M)^2$$

$$\det M = \lambda_1 \lambda_2 = AB - C^2$$

$$\text{trace} M = \lambda_1 + \lambda_2 = A + B$$

k通常取0.04~0.06

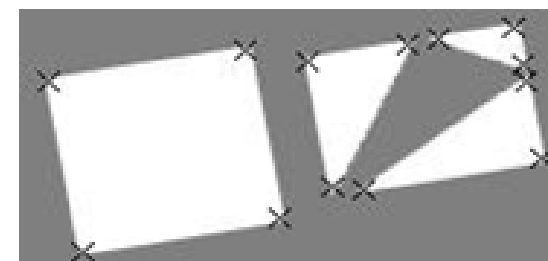
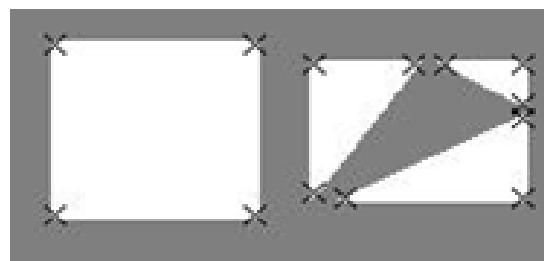
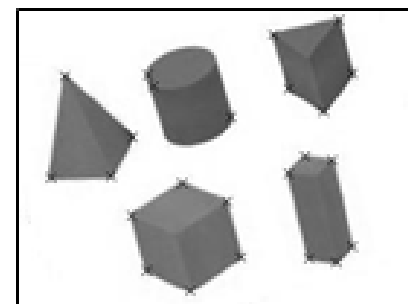
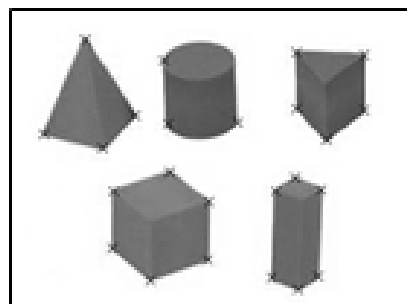
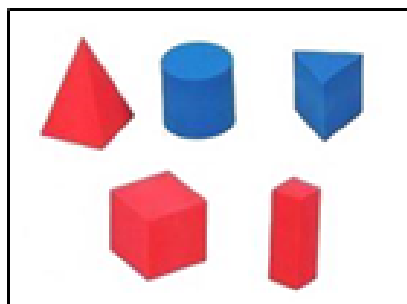
$R$ 仅由 $M$ 的特征值决定，在平坦区域绝对值较小，在边缘处为绝对值较大的负值，在角点的位置是较大的正数。因此，**当 $R$ 取局部极大值且大于给定阈值 $T$ 时的位置就是角点。**

## ■ Harris角点检测

### □ 算法

- ◆ 计算图像每一点水平和垂直方向梯度的平方以及水平和垂直梯度的乘积  $f_x^2$ 、 $f_y^2$ 、 $f_x f_y$
- ◆ 对3幅图像进行高斯滤波，构造自相关矩阵M
- ◆ 计算角点响应函数R，设定阈值T，取 $R > T$ 的位置为候选角点
- ◆ 对候选角点进行局部非极大抑制，得到角点

## ■ Harris角点检测



测试图

Harris角点检测

旋转10° 检测

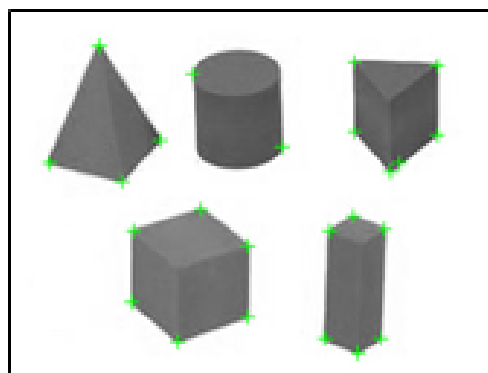
Harris角点具有旋转变换不变性，对亮度和对比度变化不敏感，不具有尺度变换不变性

## ■ 最小特征值角点检测

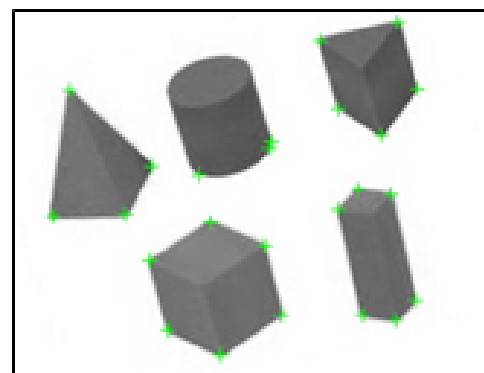
最小特征值算法 (Minimum Eigenvalue Algorithm)，修改了Harris角点检测的CRF

$$R = \min(\lambda_1, \lambda_2)$$

根据Harris角点检测中的分析，角点处 $\lambda_1$   $\lambda_2$ 取值均较大， $R$ 取值也大，选若干最大 $R$ 对应的点作为角点，计算便捷，效率较高。



旋转前检测



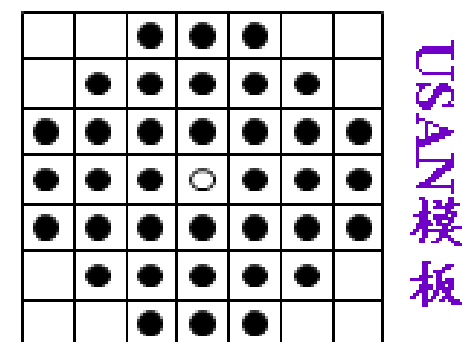
旋转后检测



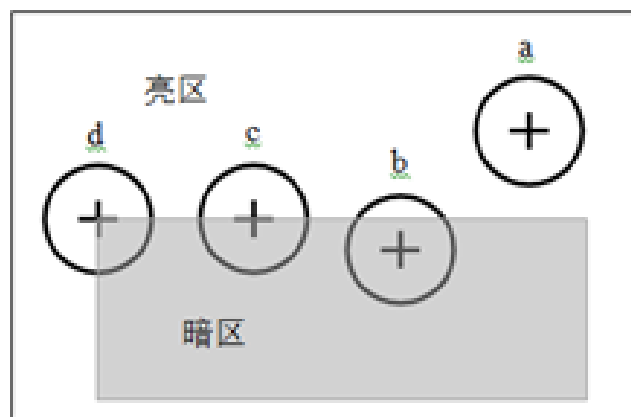
## ■ SUSAN角点检测

### □ 原理

设计USAN模板，将模板内每个像素点的灰度值和中心像素点作比较，与中心点灰度值相近的点构成的USAN区域



USAN区域  
不同位置的



USAN区域大小反映图像局部特征的强度，面积越小，该点是角点的可能性越大，通过计算比较USAN面积实现角点检测

## ■ SUSAN角点检测

### □ 算法步骤

- ◆ 将圆形模板的中心放在待测图像的像素上，计算模板内的像素与中心像素的灰度差值，统计灰度差值小于阈值T的像素个数（USAN区域面积）

$$c(r, r_0) = \begin{cases} 1 & |f(r) - f(r_0)| \leq T \\ 0 & |f(r) - f(r_0)| > T \end{cases} \quad \text{或} \quad c(r, r_0) = e^{-\left(\frac{f(r) - f(r_0)}{T}\right)^2}$$

$$n(r_0) = \sum_{r \in D(r_0)} c(r, r_0)$$

## ■ SUSAN角点检测

### □ 算法步骤

#### ◆ 计算角点响应函数值

$$R(r_0) = \begin{cases} g - n(r_0) & n(r_0) < g \\ 0 & n(r_0) \geq g \end{cases}$$

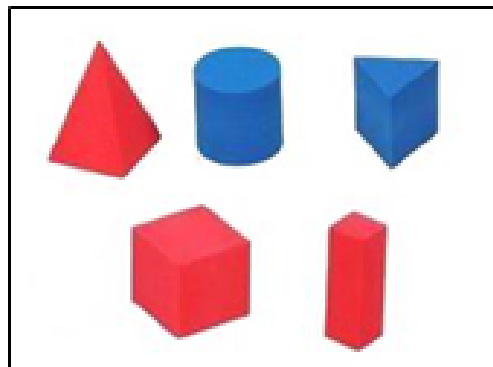
检测角点时， $g$ 设为USAN的最大面积的一半；  
检测边缘点时， $g$ 设为USAN的最大面积的3/4。

#### ◆ 排除伪角点:计算USAN的重心、重心同模板中心的距离，如果距离较小则不是正确的角点。

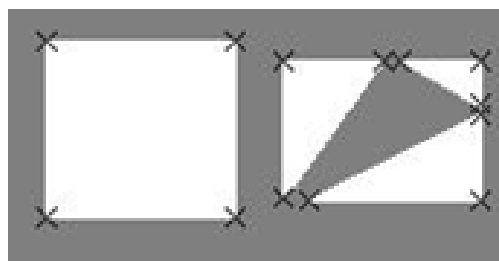
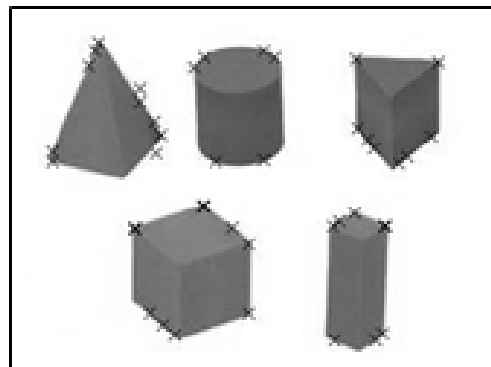
$$C_{r_0} = \frac{\sum_r r c(r, r_0)}{\sum_r c(r, r_0)}$$

#### ◆ 进行非极大抑制来求得最后的角点

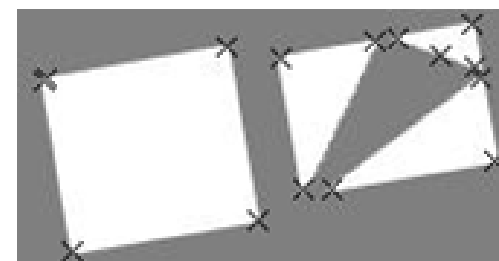
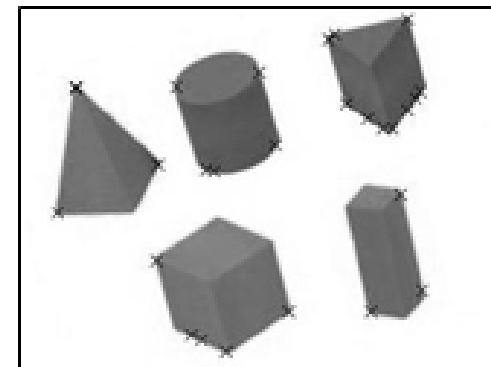
## ■ SUSAN角点检测



测试图



SUSAN角点检测



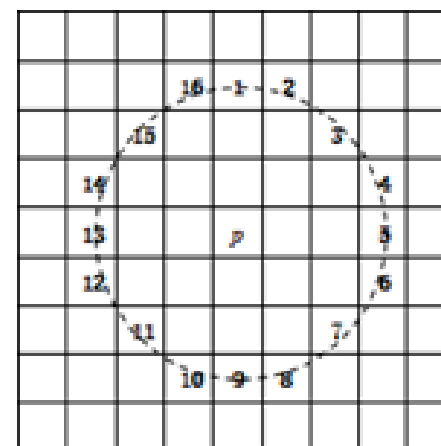
旋转后检测

理论上圆形的SUSAN模板具有各向同性，可以抵抗图像的旋转变换；算法中阈值的选择，会对程序运行结果有一定影响。

## ■ FAST角点检测

FAST (Features from Accelerated Segment Test) : 若某像素与其周围邻域内足够多的像素点相差较大, 则该像素可能是角点

- ◆ 以像素 $p$ 为中心、以3.4为半径确定一个圆, 圆上16个像素点 $p_i, i=1,2,\dots,16$ ,
- ◆ 候选角点检测。定义一个阈值 $T$ , 若 $p_i$ 中有至少 $n$ 个连续的像素点的像素值都大于 $f_p + T$ , 或者都小于 $f_p - T$ , 那么 $p$ 是一个候选角点。



阈值 $T$ 可以根据相对于 $f_p$ 的比例确定, 也可以设定为一个具体的灰度值。为提高速度, 可以仅检测上下左右四个点, 若至少有3个满足条件, 再对圆上所有点进行检测。

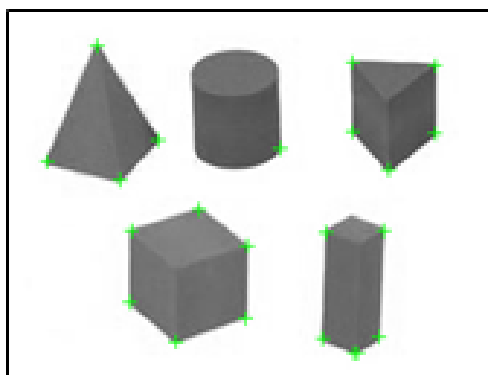
## ■ FAST角点检测

- ◆ 非极大抑制。以候选角点 $p$ 为中心的一个邻域内，若有多个候选点，则判断每个候选点的响应值 $V$ ，若 $p$ 是邻域所有候选点中响应值最大的，则保留；否则，去掉该候选点。若邻域内只有一个候选点，则保留。

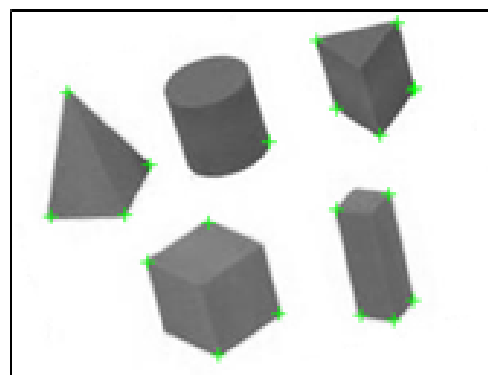
$$V = \max \left[ \sum_{f_{p_i} - f_p > T} (f_{p_i} - f_p), \sum_{f_p - f_{p_i} > T} (f_p - f_{p_i}) \right]$$

另外也设计了基于机器学习的FAST算法，需要使用ID3算法建立决策树

旋转前检测



旋转后检测



### 3.2.2 点特征描述

对于检测出的特征点，需要使用合适的特征描述方法描述特征点附近的**局部图像模式**，如梯度直方图，局部随机二值特征等。

#### ■ SIFT (Scale-invariant Feature Transform) 描述子

D.G.Lowe 1999年提出，利用金字塔和高斯核滤波差分提取局部特征，在尺度空间寻找极值点，提取位置、尺度和旋转不变量。

#### 尺度空间

将图像 $f(x,y)$ 与不同尺度因子的高斯核函数 $g(x,y,\sigma)$ 进行卷积运算，构成该图像的尺度空间 $L(x,y,\sigma)$

$$L(x,y,\sigma) = f(x,y) * g(x,y,\sigma)$$

$\sigma$ 是高斯函数的方差，取值越小，图像被平滑越少，相应的尺度也越小

## ■ SIFT描述子

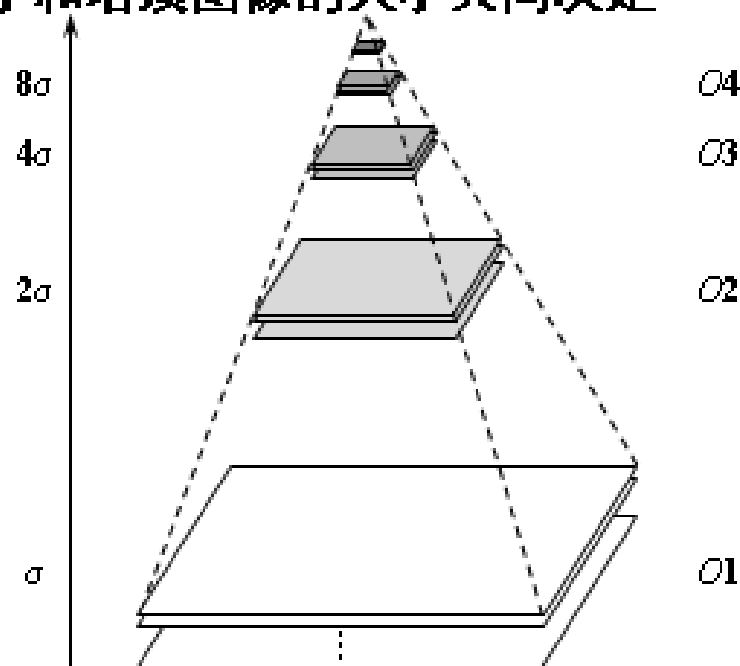
### 图像金字塔

构建一个由大到小、从下到上的**塔型图像序列**，上一层图像由下一层图像进行下采样得到。

金字塔的层数根据图像的原始大小和塔顶图像的大小共同决定

### 高斯金字塔

将每层图像使用不同参数进行高斯滤波，即金字塔的**每层含有多幅高斯模糊图像**，称为一组，组内的多幅图像按层次叠放，因此，直接称金字塔的层为组（Octave），称组内的多幅图像为层（Interval），组数为 $O$ ，每组有 $S$ 层（一般为3~5层）





## ■ SIFT描述子

### 高斯金字塔中的尺度

认为图像成像时进行了初始模糊，尺度为  $\sigma_{pre} = 0.5$ ，设置第0层尺度为  $\sigma_{init}$  (1.6)，高斯金字塔的初始尺度

$$\sigma_0 = \sqrt{\sigma_{init}^2 - \sigma_{pre}^2}$$

对于第*i*组第*j*层，尺度

$$\sigma_{i,j} = \sigma_i 2^{j/S}$$

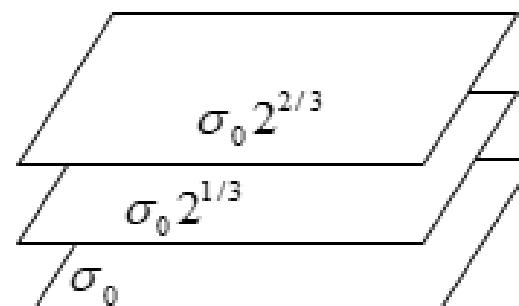
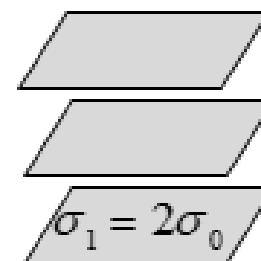
第*i*组相邻两层的尺度关系

$$\sigma_{i,j+1} = \sigma_{i,j} 2^{1/S}$$

相邻两组之间的尺度关系

$$\sigma_{i+1,j} = \sigma_i 2^{(j+S)/S} = 2\sigma_i 2^{j/S}$$

下采样时，高斯金字塔上一组图像的初始图像(底层图像)是由前一组图像的倒数第三幅图像隔点采样得到的，保证尺度连续。



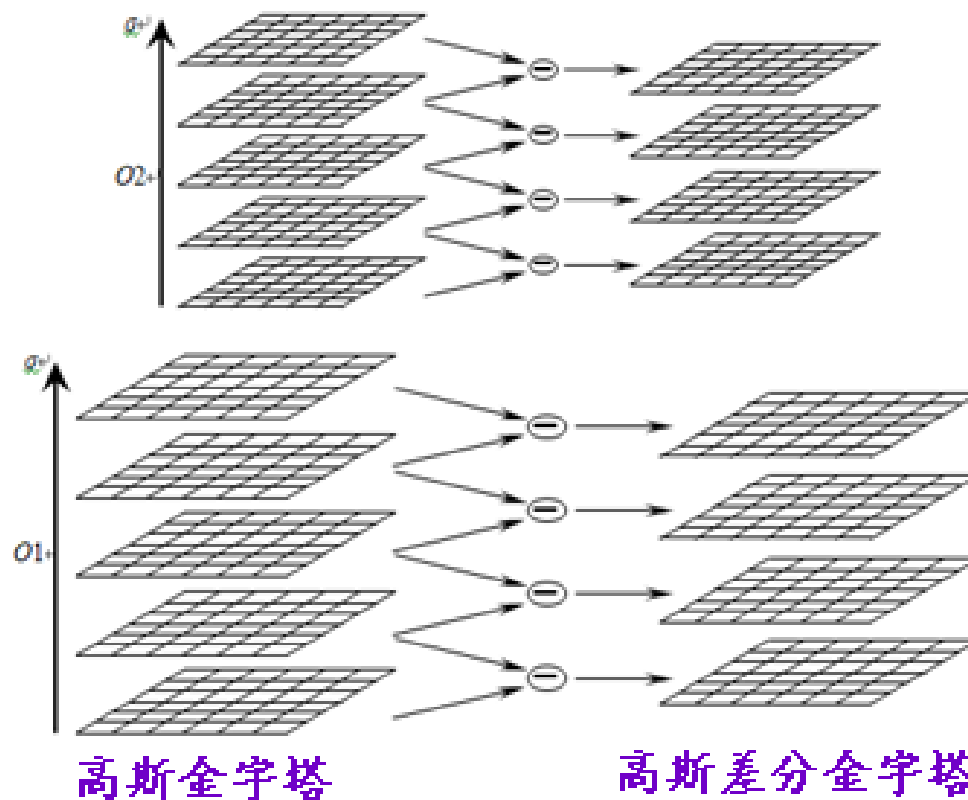
## ■ SIFT描述子

### 高斯差分尺度空间

使用高斯金字塔每组中相邻上下两层图像相减，得到高斯差分图像，简称为DoG空间

$$\begin{aligned} D(x, y, \sigma) &= L(x, y, 2^{1/5} \sigma) - L(x, y, \sigma) \\ &= [g(x, y, 2^{1/5} \sigma) - g(x, y, \sigma)] * f(x, y) \end{aligned}$$

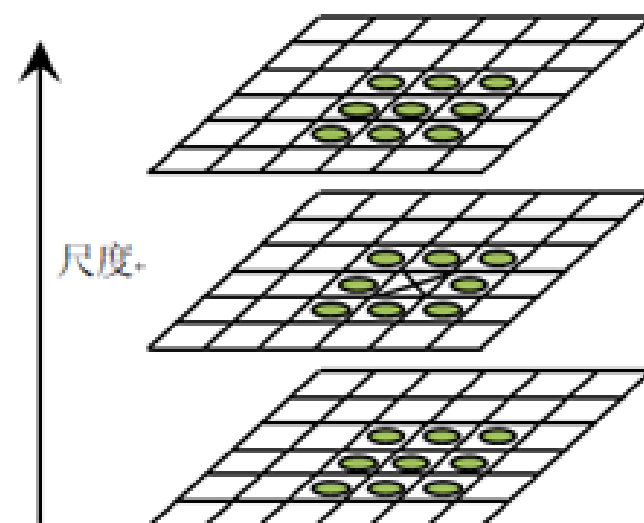
要寻找DoG空间的局部极值点



## ■ SIFT描述子

### 关键点检测

关键点由DoG空间的**局部极值点**组成的，通过**同一组内各相邻两层图像**之间比较完成：将中间层每一个像素点和同尺度的8个邻点及上下相邻尺度对应的 $9 \times 2$ 个点（共26个点）比较，若该像素点比26个邻点的DoG值都大或都小，则该点为一个局部极值。



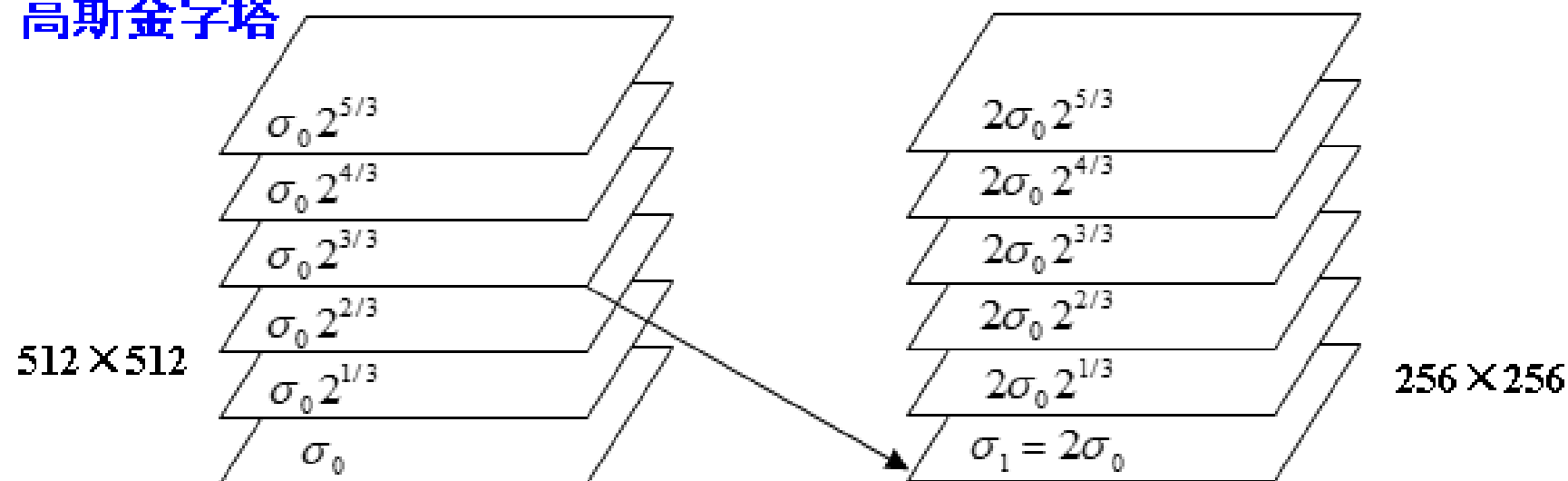
关键点检测需要前后两层高斯差分图像，所以，如果查找S层的特征点，需要 $S+2$ 层高斯差分图像，查找其中的第2层到第 $S+1$ 层。 $S+2$ 层高斯差分图像需要 $S+3$ 层高斯图像来构建出来。所以，如果整个尺度空间一共有O组，每组有 $S+3$ 层图像。

## ■ SIFT描述子

**例：**  $512 \times 512$  的图像I，构建高斯差分金字塔，每组3层

$S=3$ ，则需要高斯差分图像  $S+2=5$  幅，尺度空间图像6幅，也就是高斯金字塔每组6层。取最顶层图像大小为  $16 \times 16$ ，共  $6 \times 6=36$  幅图像

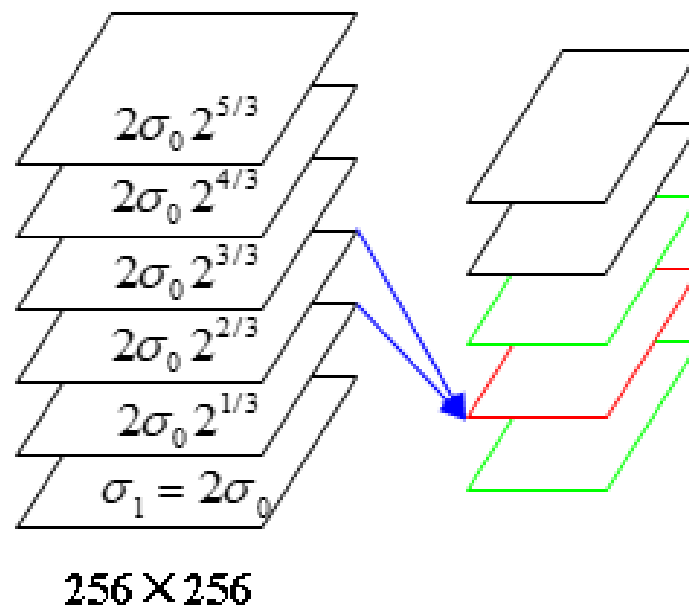
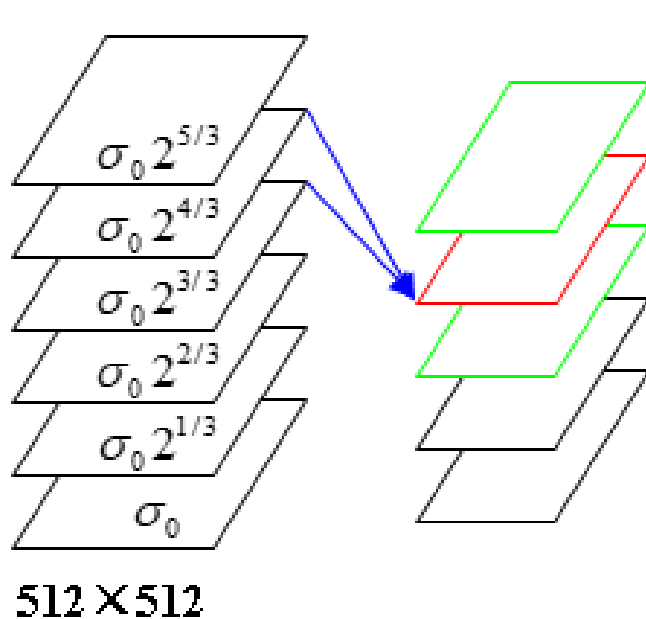
### 高斯金字塔



## ■ SIFT描述子

**例：** 512 × 512 的图像I，构建高斯差分金字塔，每组3层

### 高斯差分金字塔

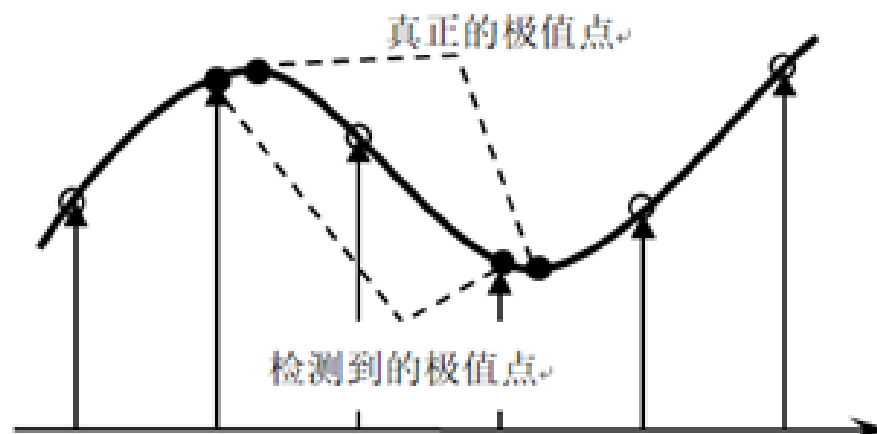


## ■ SIFT描述子

### 关键点定位

进一步去除低对比度的极值点及边缘响应点，增强匹配稳定性、提高抗噪声能力。

检测到的极值点是离散空间的极值点，与真正的极值点存在距离，影响匹配的精度，利用已知的离散空间点**插值**得到连续空间极值点。



## ■ SIFT描述子

### 关键点方向分配

为了使描述符具有旋转不变性，利用图像的局部特征为给每一个关键点指定方向参数

- ◆ 以关键点为中心，确定一个邻域，统计该邻域窗口内每一个像素点的梯度方向，生成梯度方向直方图。
- ◆ 梯度直方图将0~360度的方向范围分为36个柱，每柱10度，柱所代表的方向为像素点梯度方向，柱的长短代表了梯度幅值，峰值代表了该关键点处邻域梯度的主方向，作为该关键点的方向。
- ◆ 如果还有另一个相当于主峰值80%的峰值，认为该方向是关键点的辅方向。

## ■ SIFT描述子

### 关键点特征描述

对关键点周围图像区域分块，计算块内梯度直方图，生成具有独特性的特征向量，以便于提高特征点正确匹配的概率

- ◆ 确定计算描述子所需的图像区域：设关键点的主方向为 $\theta$ ，所在尺度为 $\sigma$ ，将关键点附近邻域划分成 $d \times d$ 个子区域，每个子区域的尺寸为 $3\sigma \times 3\sigma$ 个像素；考虑到插值需要，图像区域边长为 $3\sigma(d+1)$ ；再加上旋转因素，实际计算的图像区域半径为 $R = 3\sigma\sqrt{2}(d+1)/2$ 。
- ◆ 以关键点为中心，将附近半径为 $R$ 的圆内图像坐标旋转一个方向角 $\theta$ ，以确保旋转不变性。



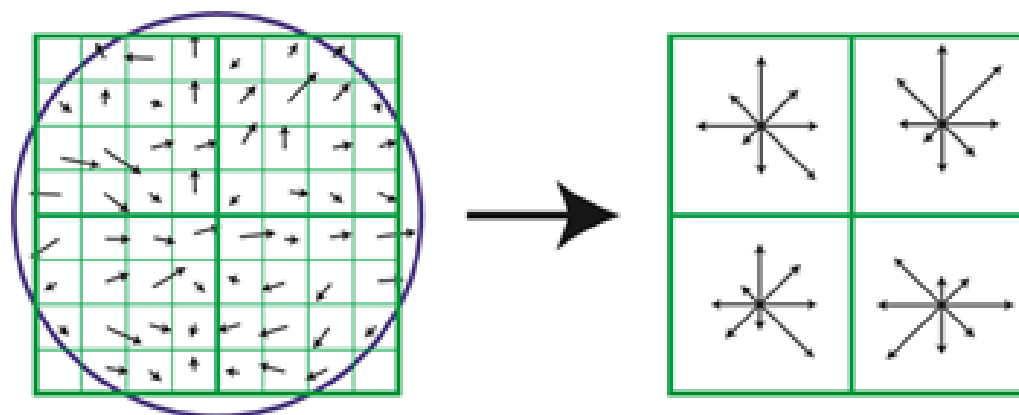
## ■ SIFT描述子

### 关键点特征描述

- ◆ 在旋转后的图像坐标下，以关键点为中心，选取 $3\sigma d \times 3\sigma d$ 大小的区域，对每个像素点计算梯度幅值和梯度方向；并对每个像素梯度幅值用尺度为 $0.5d$ 的高斯分布进行加权。
- ◆ 将 $3\sigma d \times 3\sigma d$ 大小的图像区域等间隔划分为 $d \times d$ 个子区域，在每个子区域内计算8个方向的梯度方向直方图，绘制每个梯度方向的累加值，形成 $d \times d \times 8$ 的特征向量。
- ◆ 对特征向量进行归一化处理，去除光照变化的影响。
- ◆ 设置门限值（一般取0.2）截断较大的梯度值。然后，再进行一次归一化处理，提高特征的鉴别性。
- ◆ 按特征点的尺度对特征描述向量进行排序，生成SIFT特征描述子。

## ■ SIFT描述子

### 关键点特征描述



将关键点周围选择了 $2 \times 2$ 个小区域，每个区域计算8个方向的梯度方向直方图，最终形成32维的特征向量。建议选择 $4 \times 4$ 个小区域，构成128维特征向量。

## ■ SURF (Speeded Up Robust Features) 描述子

- ◆ Herbert Bay等人于2006年提出，在生成特征矢量时，利用积分图，使用快速Hessian检测子来判断尺度空间提取的关键点是否为极值点；确定每个极值点的主方向，沿主方向构造一个窗口区域，在窗口内提取特征向量，用该向量描述关键点。
- ◆ 相比SIFT算法，SURF保持了尺度不变和旋转不变的特性，速度快，鲁棒性好。

Bay H, Ess A, Tuytelaars T, et al. Speeded-up robust features[J]. Computer vision and image understanding, 2008, 110(3): 346-359.

## ■ SURF描述子

### Hessian矩阵

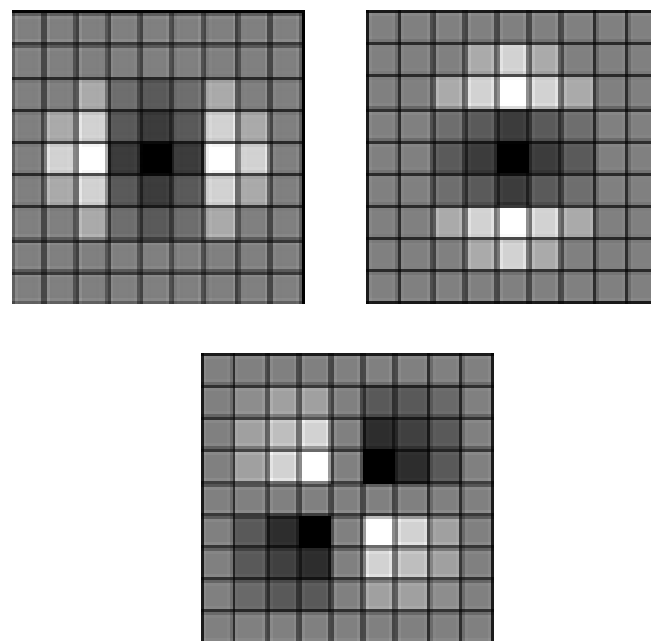
$$H(x, y, \sigma) = \begin{pmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{pmatrix}$$

**Lxx:** 图像 $f(x, y)$ 与  
高斯函数二阶偏导  
 $\frac{\partial^2 g}{\partial x^2}$ 在 $(x, y)$ 处的卷积

**Lyx:** 图像 $f(x, y)$ 与  
高斯函数二阶偏导  
 $\frac{\partial^2 g}{\partial x \partial y}$ 在 $(x, y)$ 处的卷积

**Lyy:** 图像 $f(x, y)$ 与  
高斯函数二阶偏导  
 $\frac{\partial^2 g}{\partial y^2}$ 在 $(x, y)$ 处的卷积

高斯二阶微分滤波器，取  
 $\sigma=1.2$ 、 $9 \times 9$ 的模板，作为最  
小尺度空间值对图像进行滤波



## ■ SURF描述子

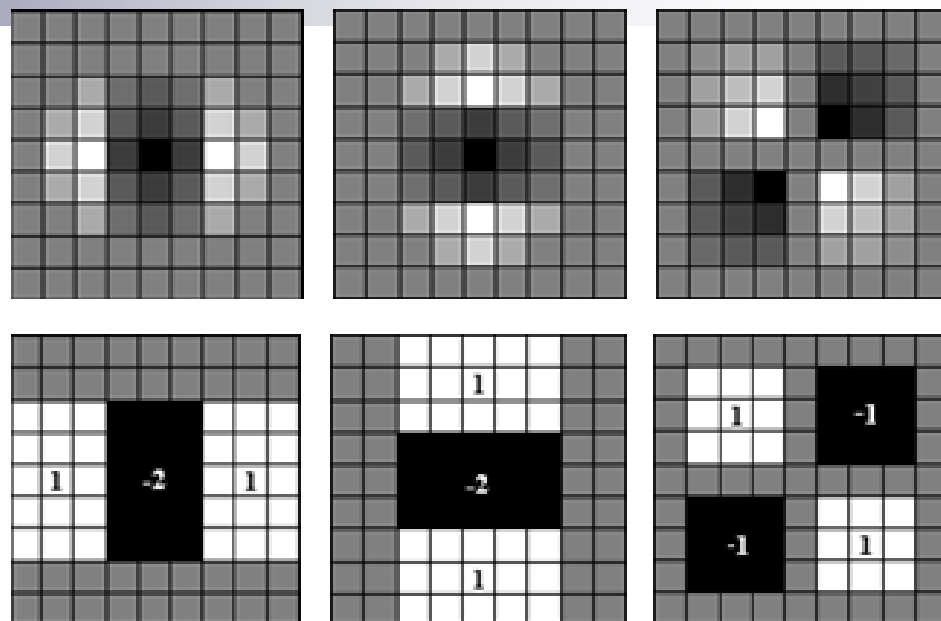
### 盒式滤波器

为**提高运算速度**，使用盒式滤波器近似替代高斯滤波器，将对图像的滤波转化为计算图像上不同区域间**像素的加减运算**问题，只需要查找**积分图**就可完成。

Hessian矩阵的行列式简化为

$$\det(H_{approx}) = D_{xx} D_{yy} - (0.9 D_{xy})^2$$

$D_{xx}$ 、 $D_{yy}$ 、 $D_{xy}$ 表示盒式滤波器与图像进行卷积的结果



图像经过盒式滤波，计算Hessian行列式的值，所有Hessian行列式值构成一幅Hessian行列式图像。

## ■ SURF描述子

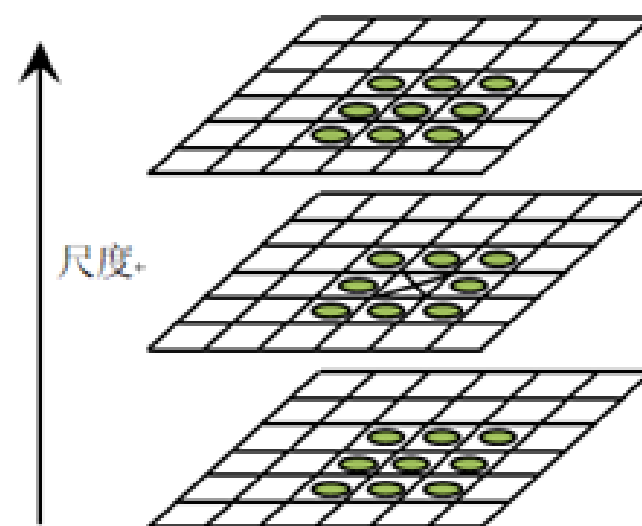
### 尺度空间的构建

- ◆ 将一幅灰度图像经过尺度空间中不同尺寸盒式滤波器的滤波处理，生成多幅Hessian行列式图像，从而构成了图像金字塔。
- ◆ 将尺度空间划分为若干组，一个组代表了逐步放大的滤波模板对同一输入图像进行滤波的一系列响应图。每组又由若干固定的层组成。两层之间的最小尺度变化量设为盒式滤波器模板尺寸的1/3。
  - ✓ 第一组： $9 \times 9$ 、 $15 \times 15$ 、 $21 \times 21$ 、 $27 \times 27$  (6)
  - ✓ 第二组： $15 \times 15$ ， $27 \times 27$ ， $39 \times 39$ ， $51 \times 51$  (12)
  - ✓ 第三组： $27 \times 27$ ， $51 \times 51$ ， $75 \times 75$ ， $99 \times 99$  (24)
  - ✓ 第四组： $51 \times 51$ ， $99 \times 99$ ， $147 \times 147$ ， $195 \times 195$  (48)

## ■ SURF描述子

### 兴趣点的检测与定位

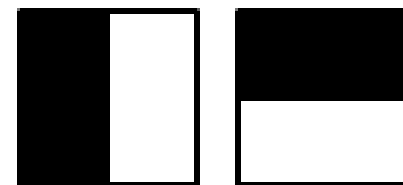
- ◆ 对计算出的行列式值设一个阈值，大于该阈值的为候选兴趣点
- ◆ 采用 $3 \times 3 \times 3$ 邻域非极大值抑制：即比较候选兴趣点与周围8个邻点及上下两层相应位置 $9 \times 2$ 个点（26个点）的行列式值，若该点行列式的值比周围26个点的值都大，则确定该点为该区域的特征点。



## ■ SURF描述子

### 兴趣点方向的分配

- ◆ 以某个兴趣点为圆心，确定以 $6s$ （ $s$ 为兴趣点的尺度）为半径的圆，用尺寸为 $4s$ 的Haar小波模板对图像进行处理，求 $x$ 、 $y$ 方向的Haar小波响应 $d_x$ 和 $d_y$



- ◆ 在圆内选择 $60^\circ$ 的扇形区域 $w$ ，统计其中的Haar小波特征总和及方向

$$m_w = \sum_w d_x + \sum_w d_y \quad \theta_w = \arctan \left( \sum_w d_y / \sum_w d_x \right)$$

- ◆ 以一定角度间隔旋转扇形区域，并再次统计该区域内的Haar小波特征值，最后将值最大的那个扇形的方向作为该特征点的主方向。

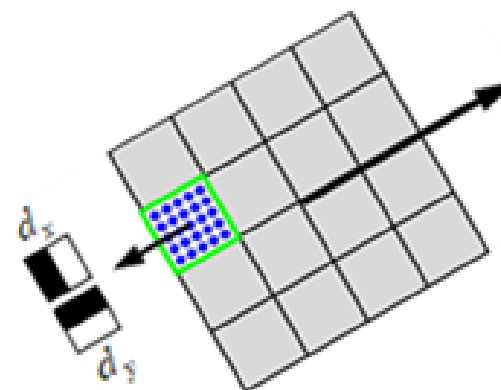


## ■ SURF描述子

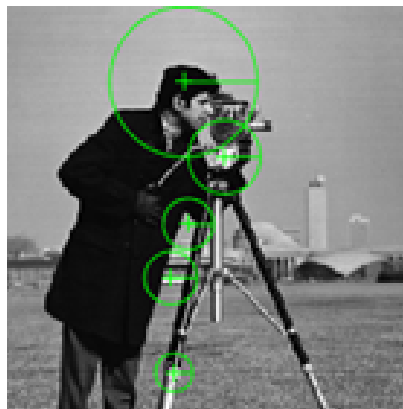
### 特征描述子的生成

- ◆ 以兴趣点为中心，沿主方向方位，构建一个大小为 $20s$ 的方形区域
- ◆ 将其划分为 $4 \times 4$ 个矩形区域，边长为 $5s$
- ◆ 对每个子区域，统计 $5 \times 5$ 个等间距采样点水平和垂直方向的Haar小波特征： $\sum dx$ 、 $\sum dy$ 、 $\sum |dx|$ 、 $\sum |dy|$ ，作为每个子块区域的特征向量
- ◆ SURF特征的描述子为 $4 \times 4 \times 4 = 64$ 维的向量

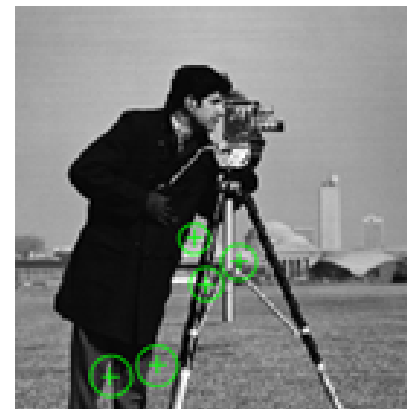
$$V = \left( \sum d_x, \sum |d_x|, \sum d_y, \sum |d_y| \right)$$



## ■ SURF描述子



5个最突出的特征点及其方向



最后5个特征点

## ■ BRISK (Binary Robust Invariant Scalable Keypoints) 描述

一种二进制的特征描述算子，通过构造图像金字塔，在尺度空间利用FAST9-16进行特征点检测，以满足尺度不变性

Leutenegger, S., M. Chli and R. Siegwart. BRISK: Binary Robust Invariant Scalable Keypoints[C]. Proceedings of the IEEE International Conference, ICCV, 2011:2548-2555.

## ■ FREAK (Fast Retina Keypoint) 描述

一种二进制的特征描述算子，通过模仿人眼视觉系统而完成

Alahi, Alexandre, Ortiz, Raphael, and Pierre Vandergheynst, FREAK: Fast Retina Keypoint[C].IEEE Conference on Computer Vision and Pattern Recognition, 2012:510-517.