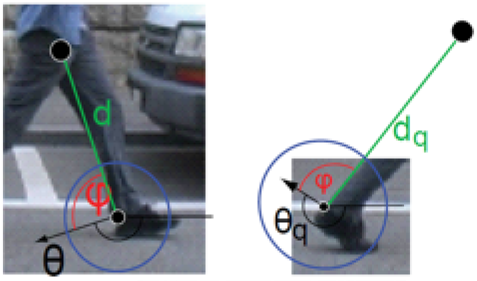
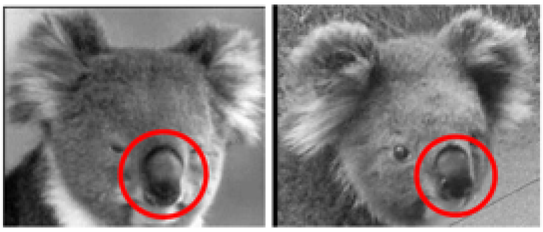
局部特征

- 全局特征会面临难以克服的困难
  - 遮挡，形变，环境

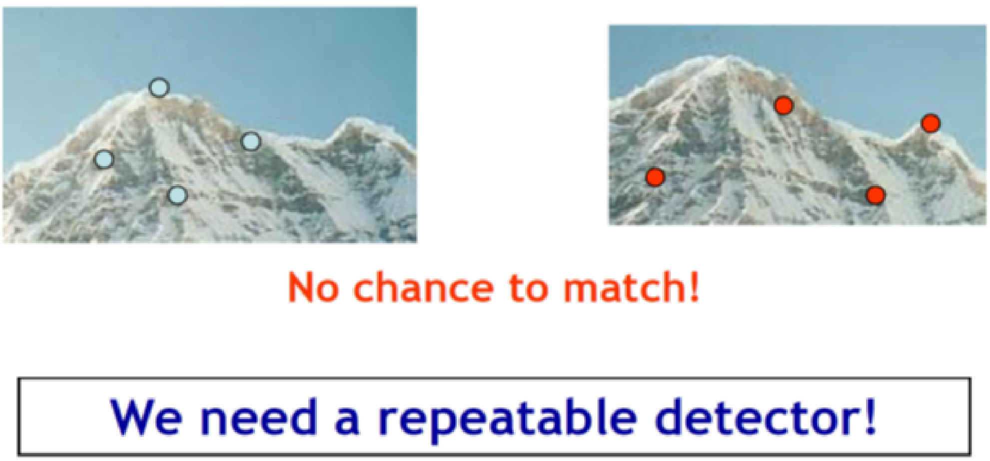




特征检测

目标：

- 稳定检测（如果两次检测出的特征点差别很大，无法进行匹配）



- 易于匹配

挑战

- 对图像几何变换的稳定性
- 对颜色，光照变化的稳定性

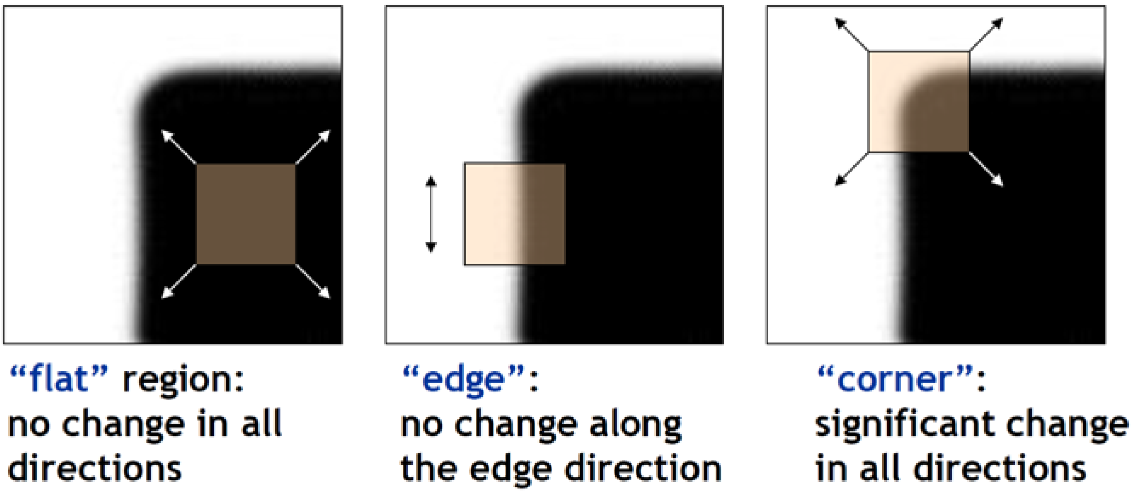
需求

- 区域提取需要是可重复的和准确的
    - 不受平移，旋转，大小的影响
    - 不受仿射变换的影响
    - 对光照变化、噪音、模糊、量有很好的适应性
- 局部性：特征是局部的，应该对遮挡和杂乱具有鲁棒性
- 数量：需要足够数量的区域来覆盖该物体
- 区别性：这些区域应该包含有趣的结构
- 效率：接近实时

角点检测

角点的概念

- 平坦区域：各方向都没有变化
- 边：单方向有变化
- 角点：各方向都有变化



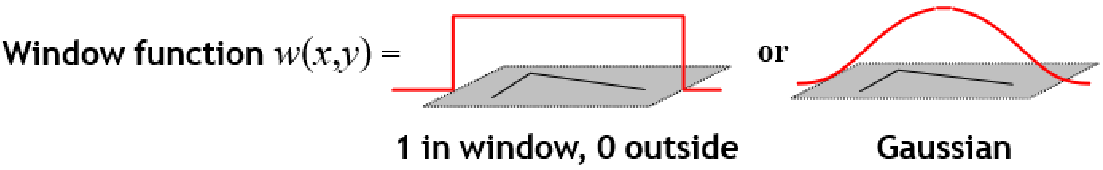
Harris角点检测

在角点处，沿任意方向运动都会引起像素颜色的明显变化

等价于：在角点附近，图像梯度具有至少两个主方向

- 数学模型：若窗口发生小位移移动 u，v，计算窗口内像素变化
  - $w(x,y)$是权重，可以设置窗口内所有位置的权重都为1，也可以利用高斯函数（角点中心的点的贡献更大）
  - $I(x+u,y+v)$ 为位置 x,y 移动后位置的像素值
  - $I(x,y)$为位置 x,y 位置的像素值

$$E(u,v)=\sum_{x,y}\underbrace{w(x,y)}_{\text{Window function}}[\underbrace{I(x+u,y+v)}_{\text{Shifted intensity}}-\underbrace{I(x,y)}_{\text{Intensity}}]^2$$



  - 将上式进行泰勒展开，可以得到如下形式

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

M 是一个 2 * 2的矩阵，元素是图像梯度

    - I_x 是 x 方向的梯度
    - I_y 是 y 方向的梯度

$$M=\underbrace{\sum_{x,y}}_{\text{Sum over image region - the area we are checking for corner}} w(x,y)\begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix}$$

Gradient with respect to x, times gradient with respect to y

$$M=\begin{bmatrix} \sum I_xI_x & \sum I_xI_y \\ \sum I_xI_y & \sum I_yI_y \end{bmatrix}=\sum\begin{bmatrix} I_x \\ I_y \end{bmatrix}[I_x \ I_y]$$

    - 若**角点是与坐标轴对齐的**，则每个点的梯度是与 x 轴平行或 与 y 轴平行的。

    如果两个 λ 都接近 0，则不是角点。角点位置的两个 λ 应该都比较大

$$M=\begin{bmatrix} \sum I_x^2 & \sum I_xI_y \\ \sum I_xI_y & \sum I_y^2 \end{bmatrix}=\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}$$

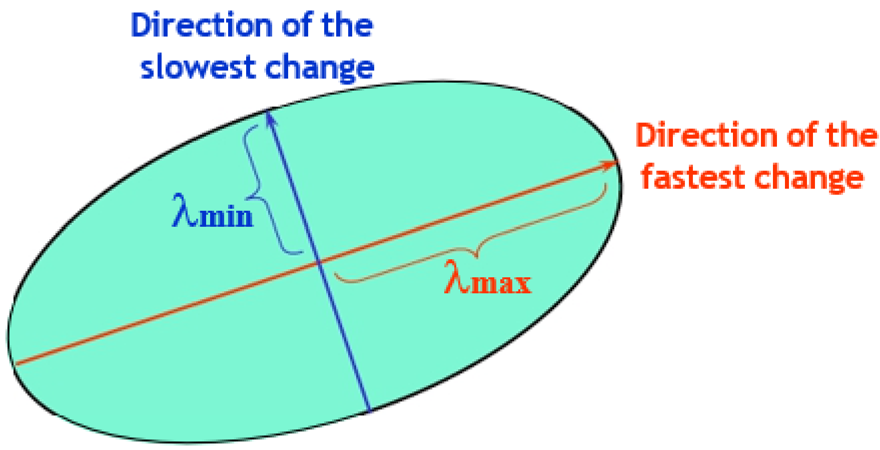
- 若**角点不与坐标轴对齐**，M 是对称的，可分解如下（对称矩阵的性质）

Since M is symmetric, we have $$M = R^{-1}\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix}R$$

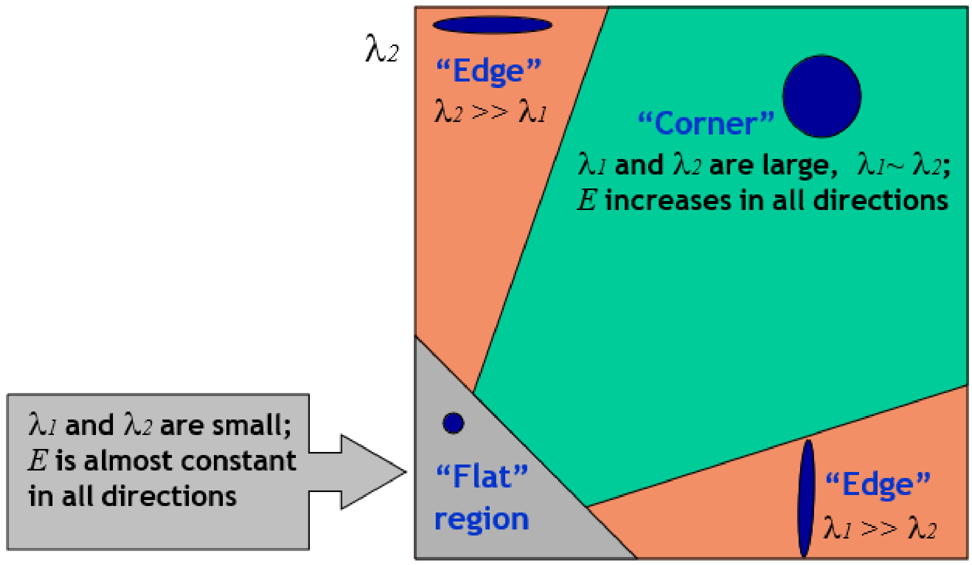
(Eigenvalue decomposition)

我们可以把M想象成一个椭圆，其轴线长度由特征值决定，方向由R决定。可以把R看成旋转因子，其不影响两个正交方向的变化分量。经对角化处理后，将两个正交方向的变化分量提取出来，就是 λ1 和 λ2（特征值）。

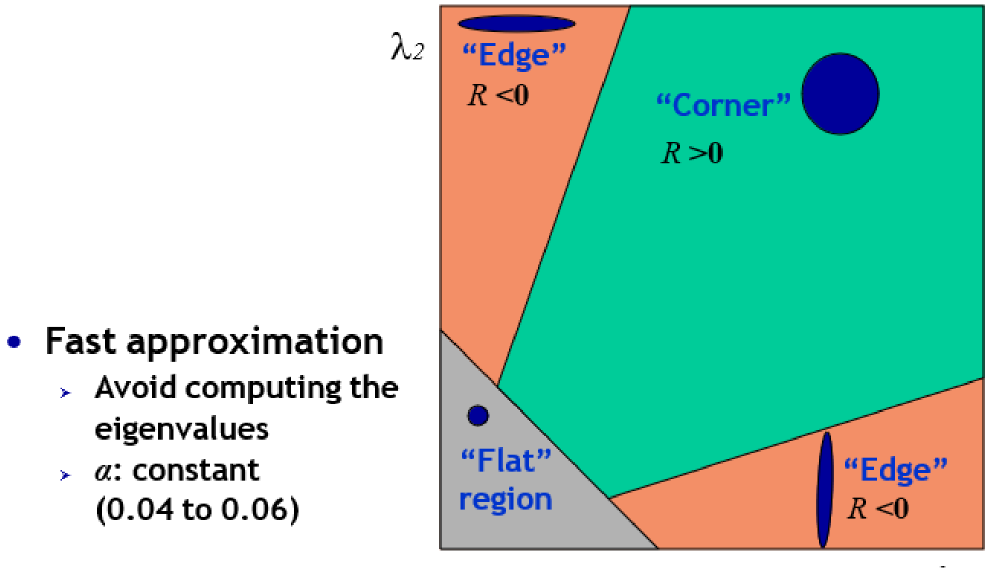
为什么可以想象成椭圆：[【特征检测】Harris角点检测中的数学推导_hujingshuang-CSDN博客]



当 λ_1 和 λ_2 都比较大，并且差不多大时，是角点



因为特征值计算起来比较麻烦，所以我们定义了响应函数
$R = det(M) - \alpha trace(M)^2 = \lambda_1\lambda_2 - \alpha(\lambda_1 + \lambda_2)^2$。角点的$|R|$大，平坦区域$|R|$小，边缘的$|R|$为负值

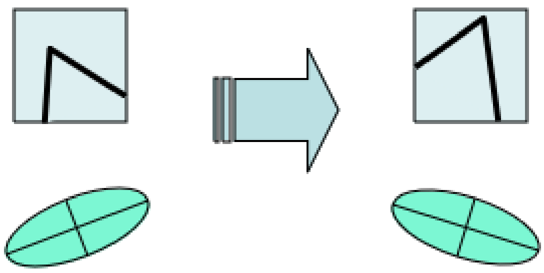


光照，几何不变性?

具有旋转不变性和光照不变性，不具有尺度不变性

- 旋转不变性

  对于下图，我们用椭圆表示 M。图像旋转后，椭圆旋转了，但是形状没变，因此响应函数 R 没变



- 不具有尺度不变性：尺度变化会将角点变成边缘
- 加法变化不变性（整体光照）：使用的是梯度

斑点检测

利用图像的$Hessian$矩阵。想法：在两个正交方向上寻找强梯度。全是二阶导

使用公式 $det(H)$，认为 $det(H)$ 大的是角点

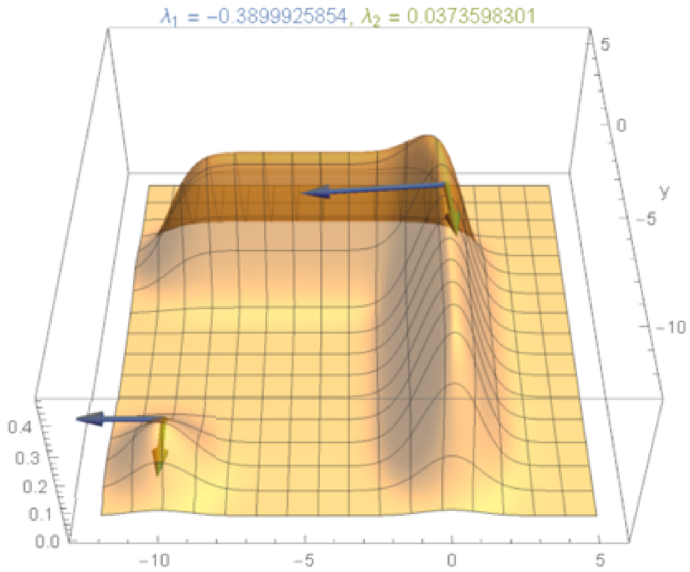
$$\det(H) = I_{xx}I_{yy} - I_{xy}^2 = \lambda_1\lambda_2$$

结果：响应主要在角点处和纹理比较强的区域，但是我们想要检测的是斑点

二阶导关注的是图像像素急剧变化的区域，同样对噪声比较敏感

$$H = \begin{pmatrix} \frac{\partial^2 I}{\partial^2 x^2} & \frac{\partial^2 I}{\partial x \partial y} \\ \frac{\partial^2 I}{\partial y \partial x} & \frac{\partial^2 I}{\partial^2 y^2} \end{pmatrix}$$

$$H = R^{-1}\begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} R$$



$Hession$对光照变化和几何变换的稳定性？

如何实现尺度不变

我们观察物体时，若物体离得近，则看起来又大又清晰。若离得远，则看起来又小又模糊。

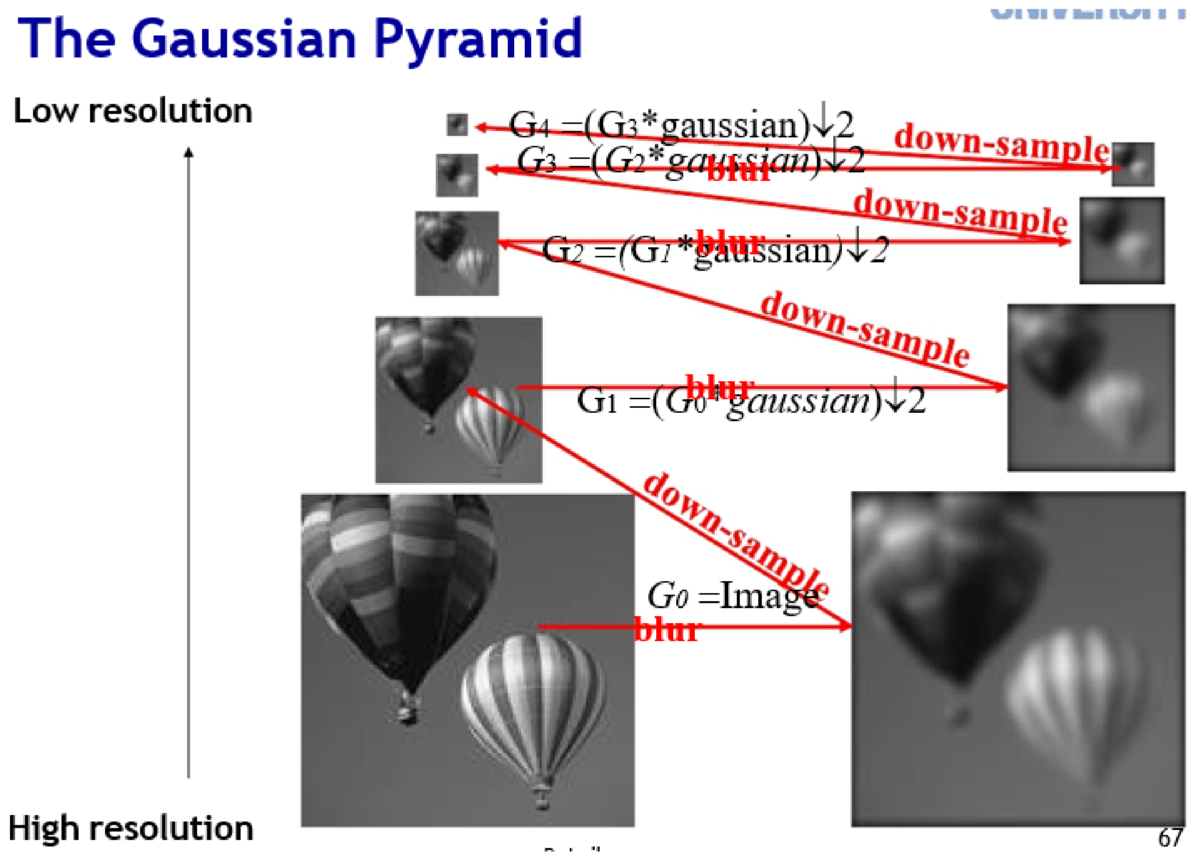
进行特征检测时，计算机并不能判断图像中物体的尺度，因此我们使用金字塔来建立一系列不同尺度的图像集

图像的尺度空间是指图像的模糊程度，而非图像的大小。近距离看一个物体和远距离看一个物体，模糊程度是不一样的；从近到远，图像越来越模糊的过程，也是图像的尺度越来越大的过程。

高斯金字塔

高斯核是实现尺度变换的唯一变换核

通过降采样构建高斯金字塔。降采样前需要先对图像进行平滑以避免伪影（信号学知识）

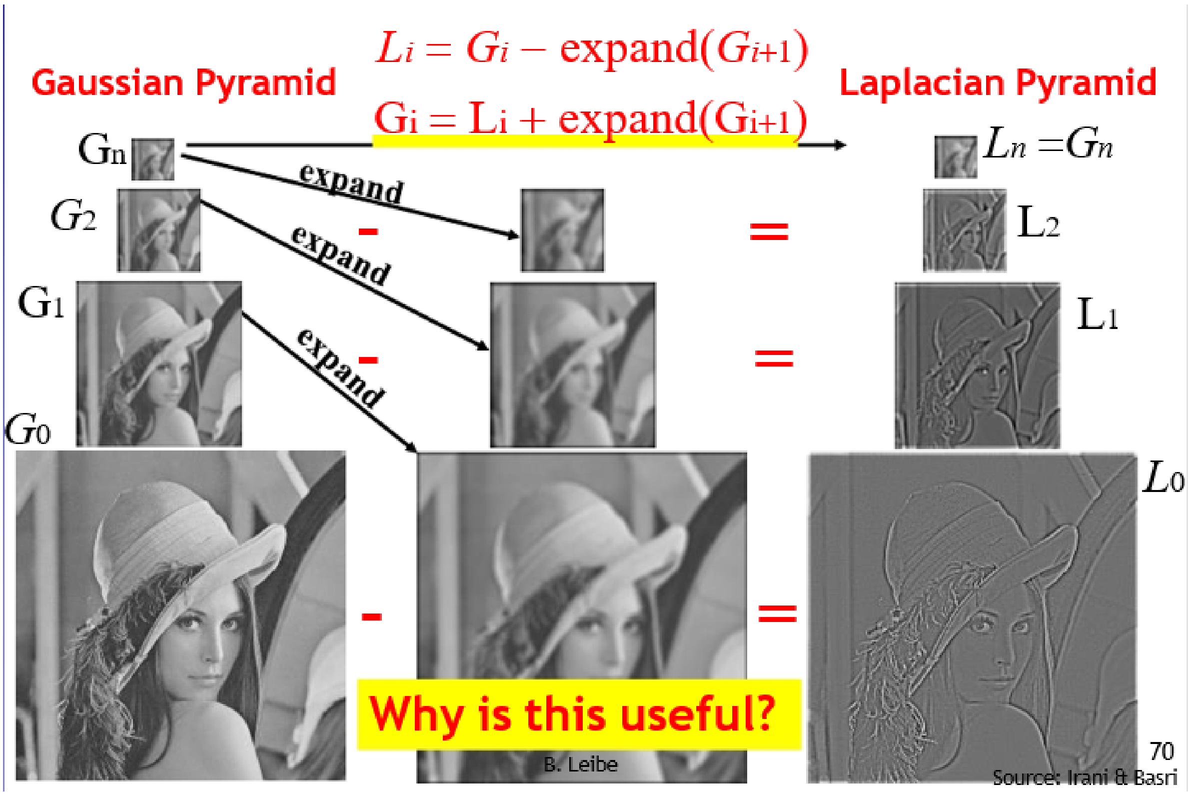


- 构建：每一层都通过对前一层进行平滑和下采样来得到
- 为什么使用高斯？部分原因：
  - 高斯 * 高斯 = 另一个高斯
  - $G(\sigma_1) * G(\sigma_2) = G(sqrt(\sigma_{12} + \sigma_{22}))$
- 高斯滤波是低通滤波，当对图像模糊后，图像中的某些信息进行融合，这些信息中是有冗余信息的因此没有必要以完整的原始分辨率来存储平滑的图像。也就是说，我们可以降采样以减少存储空间

拉普拉斯金字塔

通过用大图减去升采样(插值 + 滤波)后的小图得到。实际上进行的是 DOG 的过程（高斯的差分）

- 升采样后的图和原图大小一致，但更模糊，相当于对原图进行滤波得到的。因此两图相减是高斯的差分



- 拉普拉斯图像分解和重建
- 拉普拉斯图像融合

实现尺度不变

在不同尺度同时进行检测与匹配

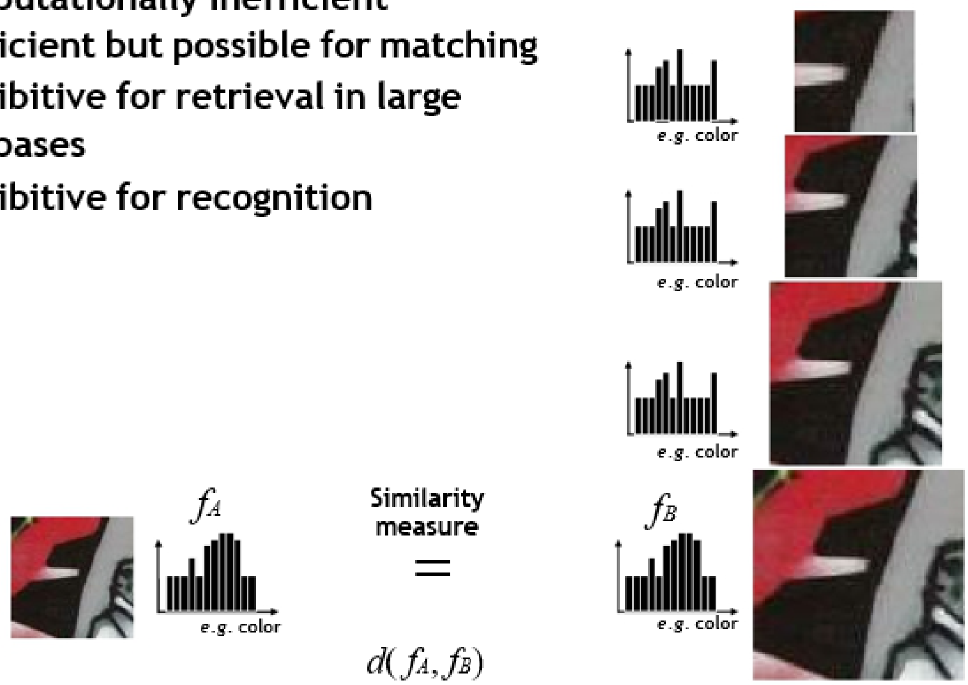
暴力

利用多个size的检测框对图像进行扫描

缺点：

- 计算效率低下
- 效率低下，但可用于匹配
- 对大型数据库的检索来说是不可能的
- 不利于识别

- Comparing descriptors while varying the patch size
  - Computationally inefficient
  - Inefficient but possible for matching
  - Prohibitive for retrieval in large databases
  - Prohibitive for recognition



自动尺寸选择

思想：针对检测区域设计一个尺度不变的函数（不管尺寸是怎样的，只要包含的内容一样，值就一样）

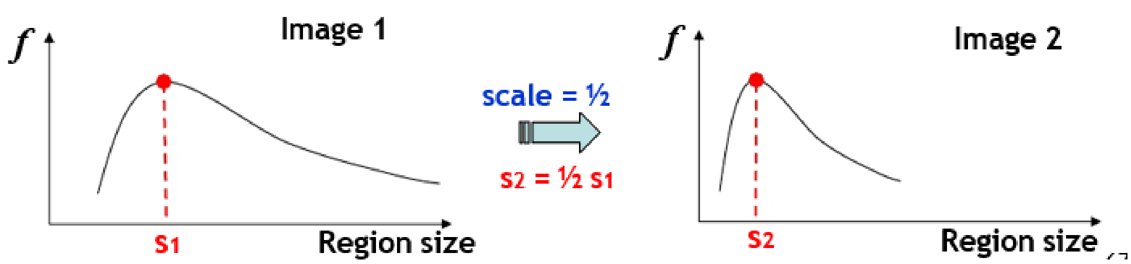
方法：

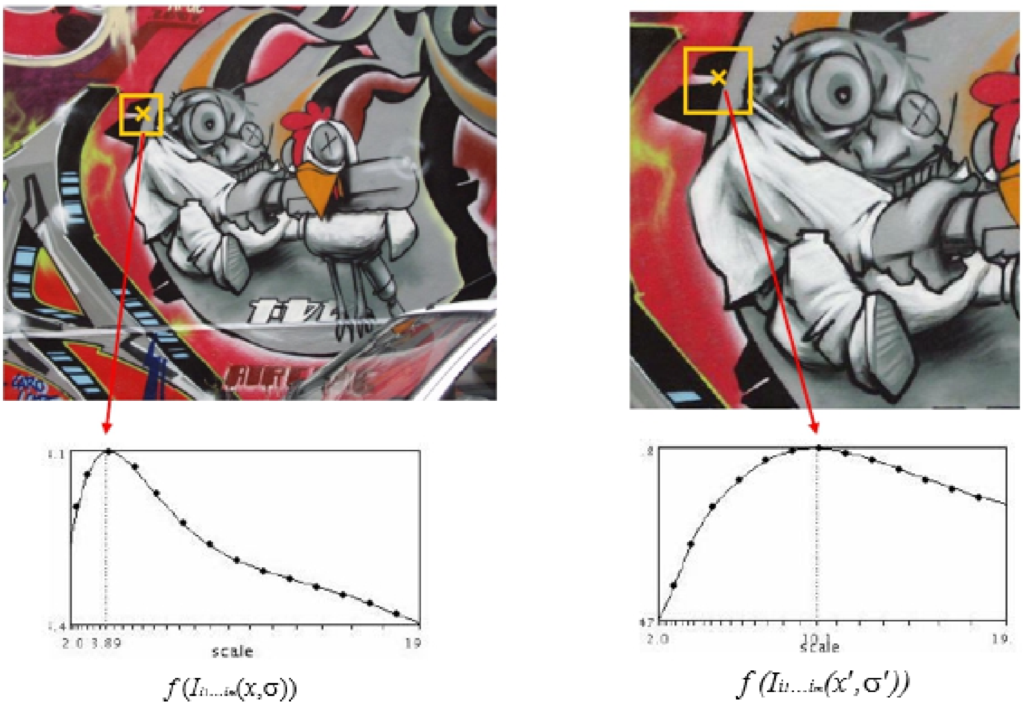
- 取函数的局部最大值

  为什么我们这里要找最大值呢？

  原因：进行特征检测时，我们往往定义一个指标，并判断这个指标最大值对应的区域为特征点
- 最大值对应的区域对图像尺寸来说是不变的

  图像尺寸变为 1/2， 则最大响应值对应的区域大小也是 1/2



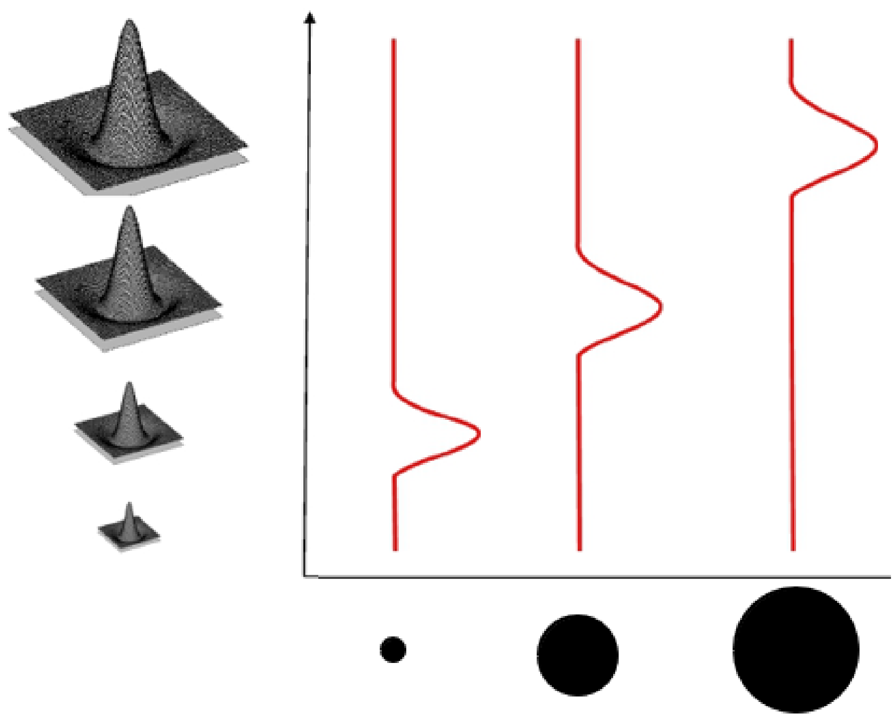


归一化：区域大小变换到固定的尺寸

什么样的函数是有效的呢?

LOG：LOG的不同 σ 影响着其检测框的scale。σ 越大，scale越大（因此可以使用不同 σ 的 LOG 来进行不同尺寸的特征检测）

- Laplacian-of-Gaussian = “blob” detector



我们记录 interest points：LOG尺度空间上的局部最大值对应的位置

实际使用时，我们常使用DOG来近似代替LOG，因为这样更方便计算

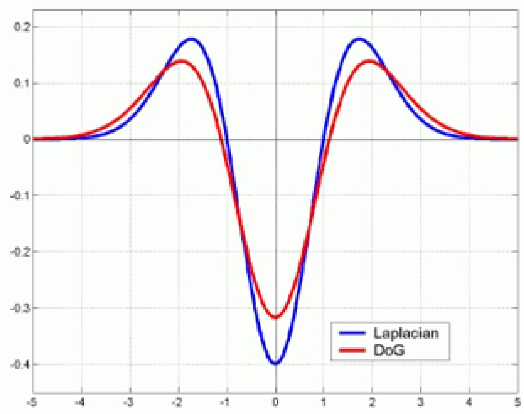
- **We can efficiently approximate the Laplacian with a difference of Gaussians:**

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

(Laplacian)

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

(Difference of Gaussians)



- 流程
  - 在尺度空间上检测 DOG 的局部最大值
  - 非极大值抑制
  - 消除边缘响应（由于某些原因，在检测极值点时，边缘容易被检测出来）