



计算机视觉

SIFT特征检测

讲师：屈老师

1. 了解SIFT变换的整体思路
2. 掌握图像的尺度空间思想
3. 了解SIFT特征检测的计算过程



- SIFT的全称是Scale Invariant Feature Transform（尺度不变特征变换），是由加拿大教授David G.Lowe在1999年在会议文章中提出，2004年发表在IJCV上
- 是计算机视觉界近二十年来引用率最高的文章之一
- SIFT特征对旋转、尺度缩放、亮度变化等保持不变性，是一种稳定的局部特征
- SIFT的特征提取方面对计算机视觉近年来的发展影响深远，特别是几乎影响到了后续所有的角点提取和匹配算法

- 图像的局部特征，对旋转、尺度缩放、亮度变化保持不变，对视角变化、仿射变换、噪声也保持一定程度的稳定性。
- 独特性好，信息量丰富，适用于海量特征库进行快速、准确的匹配。
- 多量性：即使是很少几个物体也可以产生大量的SIFT特征
- 高速性：改进的SIFT匹配算法甚至可以达到实时性
- 扩展性：可以很方便的与其他的特征向量进行联合。



SIFT算法

- 尺度空间极值检测
- 关键点定位
- 关键点特征描述

尺度空间:

- 人眼可自动调节尺度，完成对物体的检测和识别
- 模仿人的视觉认知，把物体不同尺度下的图像都提供给机器，让机器能够对物体在不同的尺度下综合信息识别
- 因此，首先需要建立尺度空间
- 通过高斯函数与原图像卷积，并经过下采样，可建立原始图像的尺度空间模型

二维高斯函数

- 高斯模糊是一种图像滤波器，它使用高斯函数计算模糊模板，并使用该模板与原图像做卷积运算，达到模糊图像的目的。

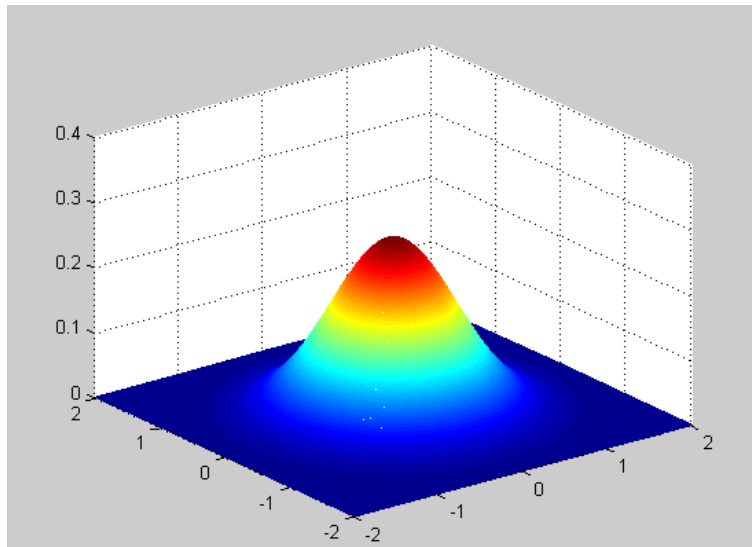
- 如二维模板大小为 $m \times n$ ，则模板上的元素 (x, y) 对应的高斯计算公式为：

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-m/2)^2 + (y-n/2)^2}{2\sigma^2}}$$

- 其中， σ 是正态分布的标准差， σ 值越大，图像越模糊(平滑)。 r 为模糊半径，模糊半径是指模板元素到模板中心的距离。

二维高斯函数

- 在二维空间中，这个公式生成的曲面的等高线是从中心开始呈正态分布的同心圆。
- 在实际应用中，在计算高斯函数的离散近似时，在大概 3σ 距离之外的像素都可以看作不起作用，这些像素的计算也就可以忽略。
- 图像处理程序只需要计算 $(6\sigma+1) \times (6\sigma+1)$ 的矩阵就可以保证相关像素影响。

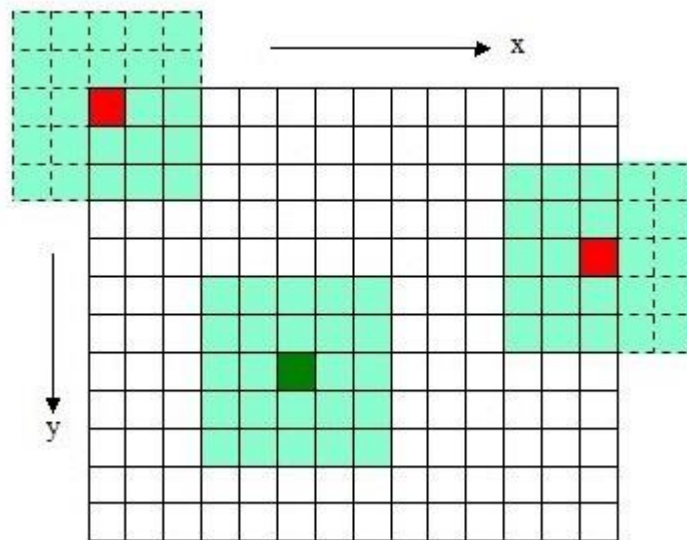


- 根据 σ 的值，计算出高斯模板矩阵的大小

$(6\sigma+1) * (6\sigma+1)$ ，使用公式 $G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-m/2)^2+(y-n/2)^2}{2\sigma^2}}$

计算高斯模板矩阵的值，与原图像做卷积，即可获得原图像的高斯模糊图像。为了确保模板矩阵中的元素在 $[0,1]$ 之间，需将模板矩阵归一化。

- 下图是 $5*5$ 的高斯模板卷积计算示意图。高斯模板是中心对称的。



- 二维高斯模糊效果图:



a) 原图



b) $\sigma = 0.6$



c) $\sigma = 10.0$



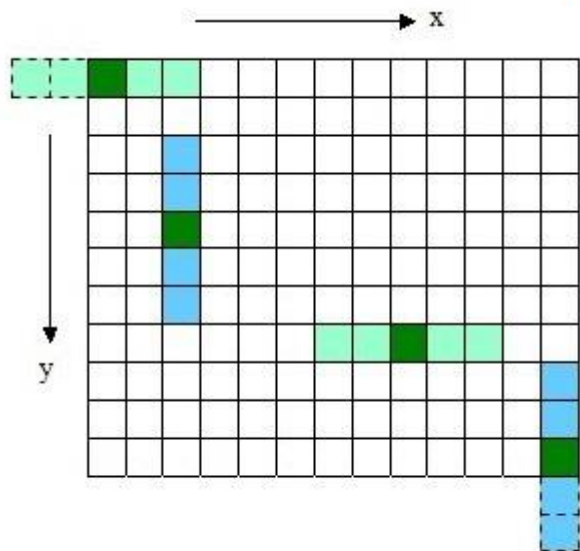
d) $\sigma = 10.0$ (边缘处理)

使用二维的高斯模板达到了模糊图像的目的，但是会因模板矩阵的关系而造成边缘图像缺失(b,c)。σ越大，缺失像素越多,丢弃模板会造成黑边(d)。更重要的是当图片变大时，高斯模板(高斯核)和卷积运算量将大幅度提高。根据高斯函数的可分离性，可对二维高斯模糊函数进行改进。

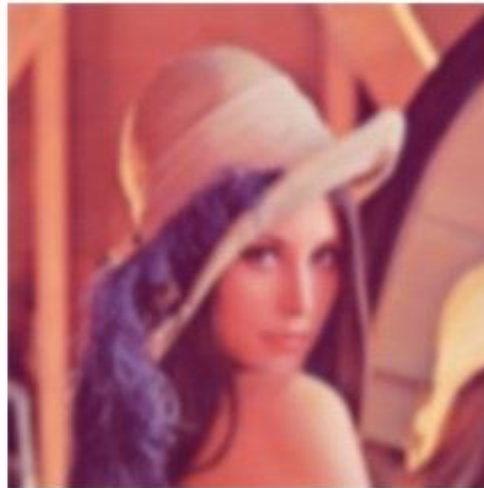
分离高斯模糊：

高斯函数的可分离性是指使用二维矩阵变换得到的效果也可以通过在水平方向进行一维高斯矩阵变换加上竖直方向的一维高斯矩阵变换得到。两次一维的高斯卷积将消除二维高斯矩阵所产生的边缘。

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-m/2)^2 + (y-n/2)^2}{2\sigma^2}}$$



a) 原图



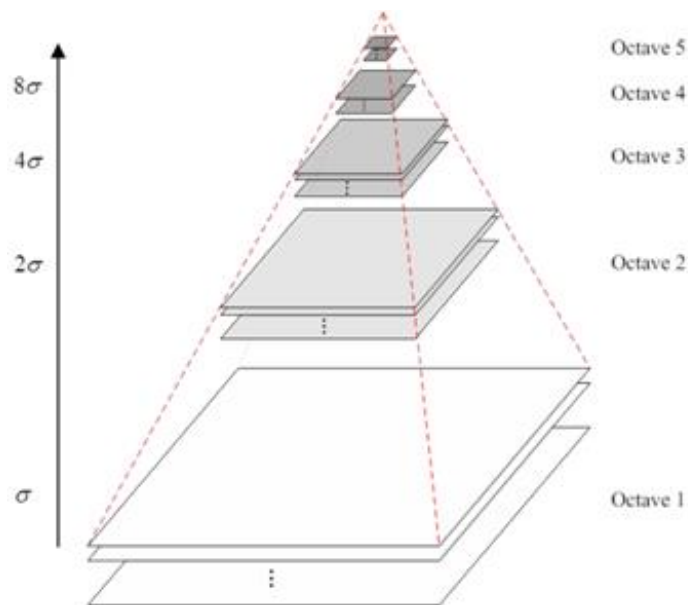
b) $\sigma = 2.0$

尺度空间理论：

- 在图像信息处理模型中引入一个被视为尺度的参数，通过连续变化尺度参数获得多尺度下的尺度空间表示序列，对这些序列进行尺度空间主轮廓的提取，并以该主轮廓作为一种特征向量，实现边缘、角点检测和不同分辨率上的特征提取等。
- 尺度空间方法将传统的单尺度图像信息处理技术纳入尺度不断变化的动态分析框架中，更容易获取图像的本质特征。尺度空间中各尺度图像的模糊程度逐渐变大，能够模拟人在距离目标由近到远时目标在视网膜上的形成过程。

高斯金字塔的构建

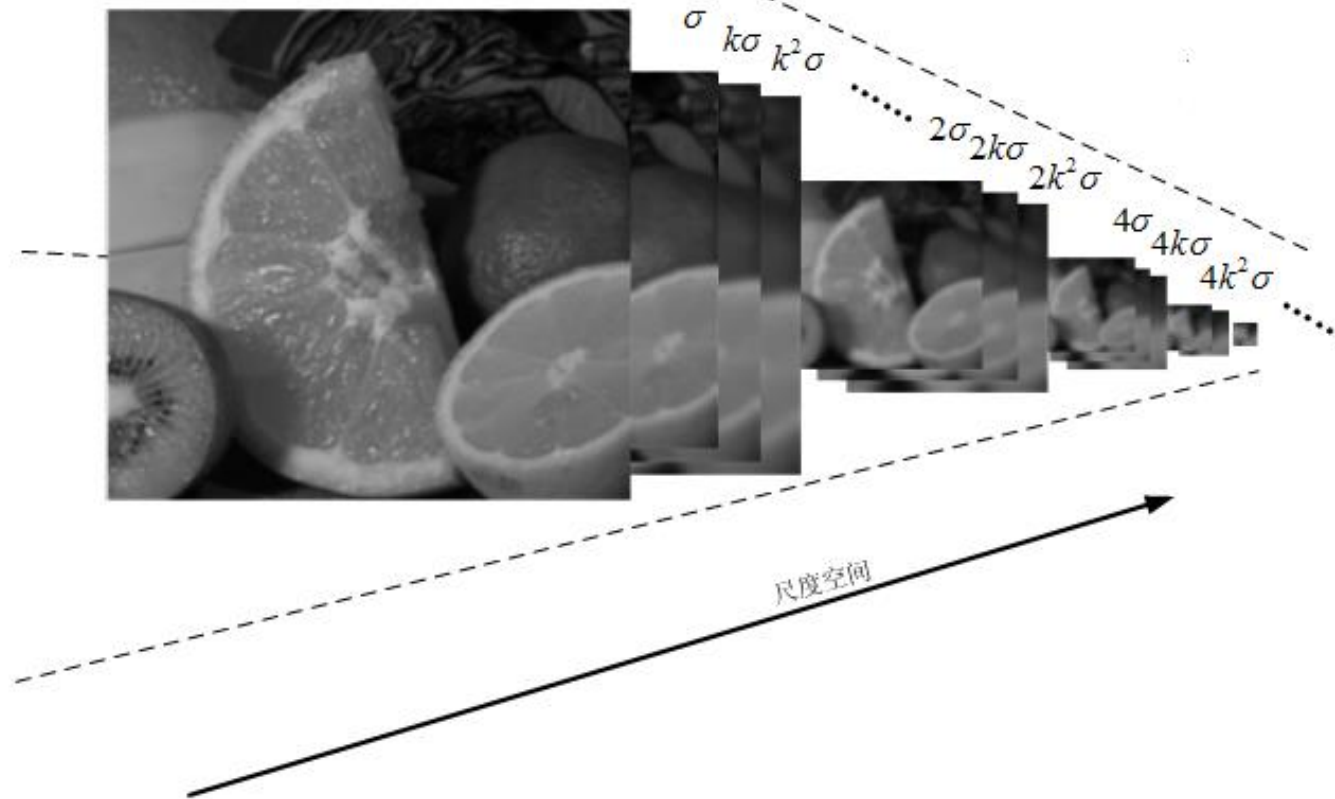
- 尺度空间在实现时使用高斯金字塔表示，高斯金字塔的构建分为两部分：
 1. 对图像做不同尺度的高斯模糊；
 2. 对图像做降采样(隔点采样)。
- 图像的金字塔模型是指，将原始图像不断降阶采样，得到一系列大小不一的图像，由大到小，从下到上构成的塔状模型。



尺度空间的形象描述

其中 $k = 2^{\frac{1}{s}}$

S 为组内总层数。

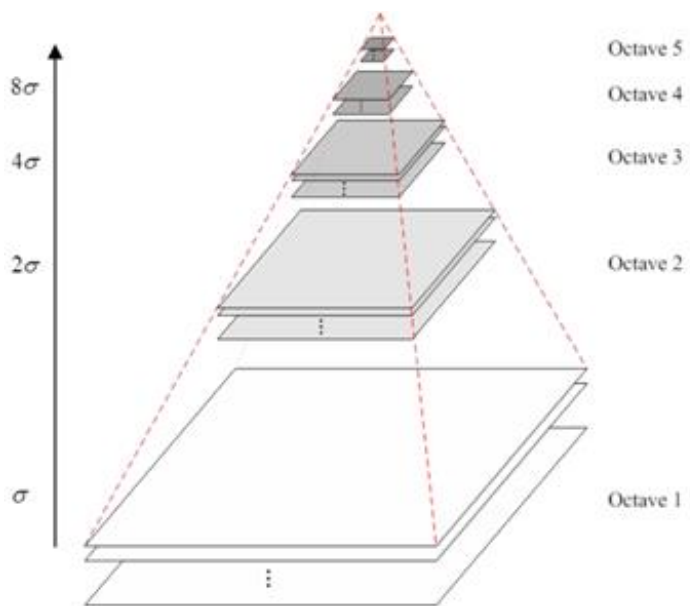


- 一个图像的尺度空间 $L(x,y,\sigma)$ ，定义为一个变化尺度的高斯函数 $G(x,y,\sigma)$ 与原图像 $I(x,y)$ 的卷积。
- $L(x,y,\sigma)=G(x,y,\sigma)*I(x,y)$ ，其中 $*$ 表示卷积运算。

$$G(x, y, \theta) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-\frac{m}{2})^2+(y-\frac{n}{2})^2}{2\sigma^2}}$$

- σ 是尺度空间因子，值越小表示图像被平滑的越少，相应的尺度也就越小。大尺度对应于图像的概貌特征，小尺度对应于图像的细节特征。

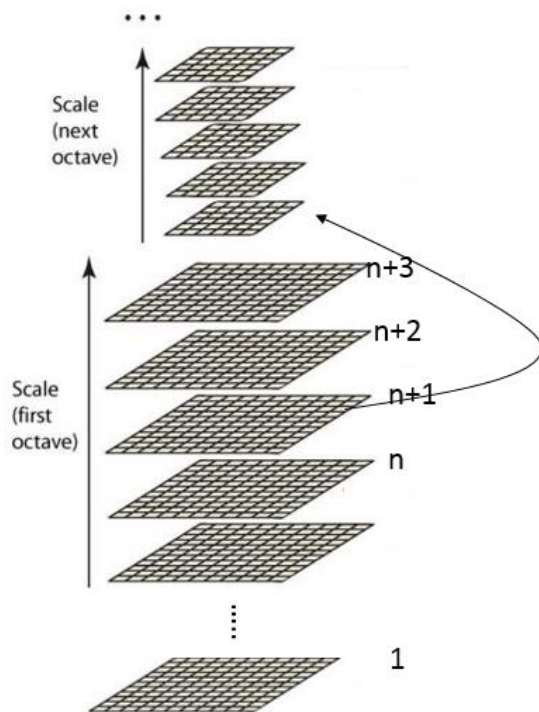
原图像为金子塔的第一层，每次降采样所得到的新图像为金字塔的一层(每层一张图像)，每个金字塔共 n 层。金字塔的层数根据图像的原始大小和塔顶图像的大小共同决定，其计算公式如下：



$$n = \log_2 \{ \min(M, N) \} - t, t \in [0, \log_2 \{ \min(M, N) \}]$$

其中 M, N 为原图像的大小, t 为塔顶图像的最小维数的对数值。如，对于大小为 512×512 的图像，当塔顶图像为 4×4 时， $n=7$ ，当塔顶图像为 2×2 时， $n=8$ 。

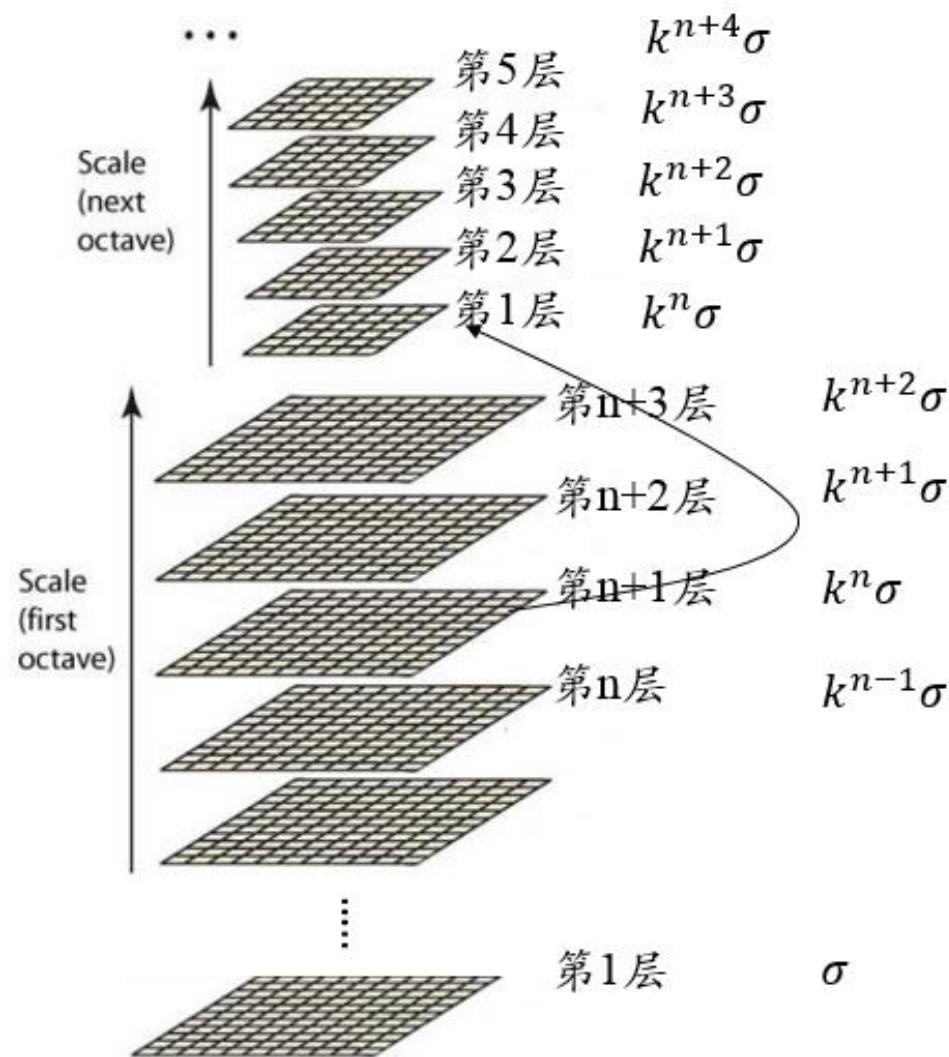
- 为了让尺度体现其连续性，高斯金字塔在简单降采样的基础上加上了高斯滤波。如下图所示，将图像金字塔每层的一张图像使用不同参数做高斯模糊，使得金字塔的每层含有多张高斯模糊图像，将金字塔每层多张图像合称为一组(Octave)，金字塔每层只有一组图像，组数和金字塔层数相等，使用公式 $n = \log_2 \{\min(M, N)\} - t, t \in [0, \log_2 \{\min(M, N)\}]$ ，计算，每组含有多层(Interval)图像。



- 在一组的图像内，每一层之间的高斯模糊的尺度因子的比值为 $k = 2^{\frac{1}{S}}$ ，其中S为每组尺度数
- 于是同一组的第s层高斯模糊尺度 $\sigma(s)$ 就变成了

$$\sigma(s) = \sigma_0 * k^{(s-1)}$$

- 这里 σ_0 是这一组的第一层图像的高斯模糊尺度。第n+1组的图像，它的高斯模糊尺度是 $2\sigma_0$ ，这个图像的高斯模糊带到了下一组的第一层图像中去，于是不同组相同层的高斯模糊尺度是2倍关系。如图所示：



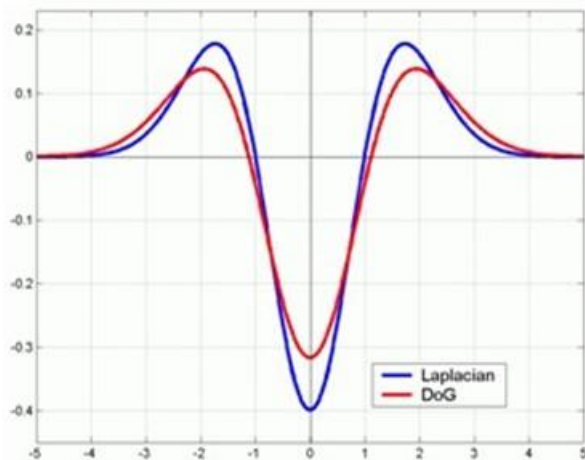
高斯差分金字塔

- 2002年Mikolajczyk在详细的实验比较中发现尺度归一化的高斯拉普拉斯函数(LoG算子 $\nabla^2 G$) $\sigma^2 \nabla^2 G$ 的极大值和极小值与其它特征提取函数如Hessian或Harris角特征相比,能够产生最稳定的图像特征。其中 $D(x,y,\sigma)$ 和 $\sigma^2 \nabla^2 G$ 的关系可以从如下公式推导得到:

- 利用差分近似代替微分, 则有: $\frac{\partial G}{\partial \sigma} = \sigma \nabla^2 G$

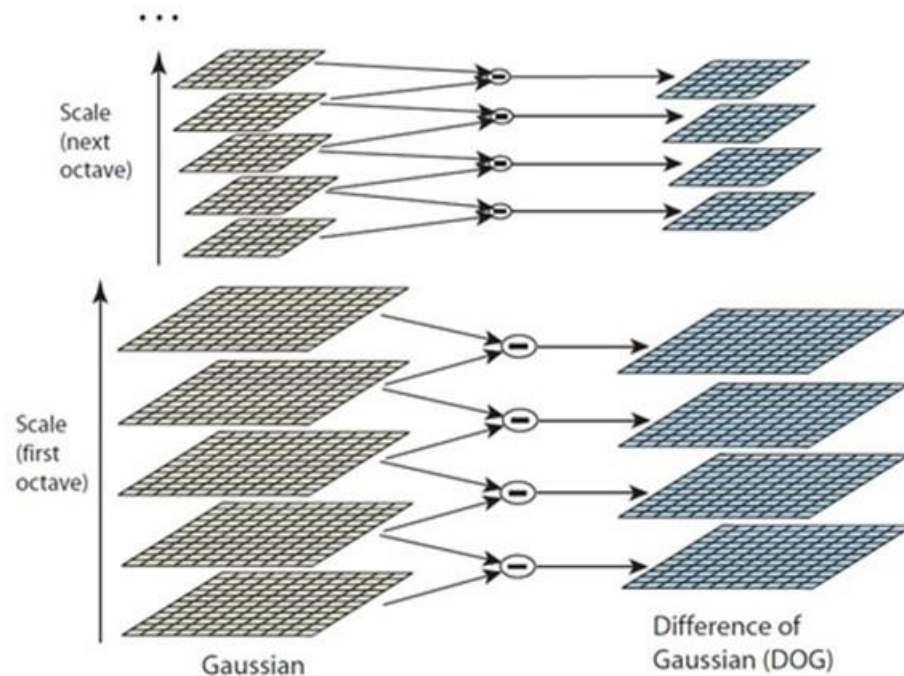
$$\sigma \nabla^2 G = \frac{\partial G}{\partial \sigma} \approx \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma}$$

- 因此 $G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2 \nabla^2 G$
- 其中 $k-1$ 是个常数, 并不影响极值点位置的求取。



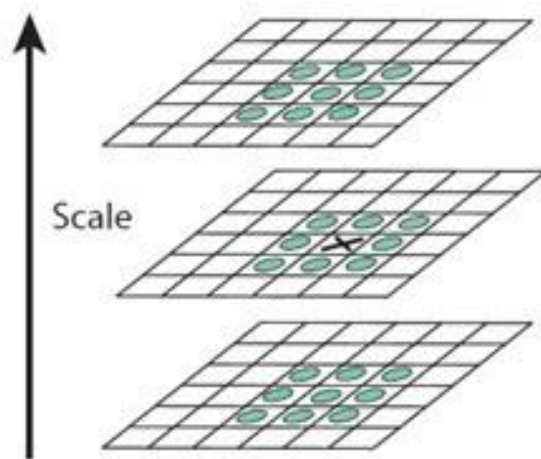
- 如图所示，红色曲线表示的是高斯差分算子(Difference of Gaussian，简称DoG)，而蓝色曲线表示的是高斯拉普拉斯(LoG)算子。使用更高效的高斯差分算子代替拉普拉斯算子进行极值检测，如下：

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



- 在实际计算时，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，如图所示，进行极值检测。这样就能得到候选的sift特征点。

- 关键点是由DoG空间的局部极值点组成的，关键点的初步探查是通过同一组内各DoG相邻两层图像之间比较完成的。为了寻找DoG函数的极值点，每一个像素点要和它所有的相邻点比较，看其是否比它的图像域和尺度域的相邻点大或者小。如图所示，中间的检测点和它同尺度的8个相邻点和上下相邻尺度对应的 9×2 个点共26个点比较，以确保在尺度空间和二维图像空间都检测到极值点。



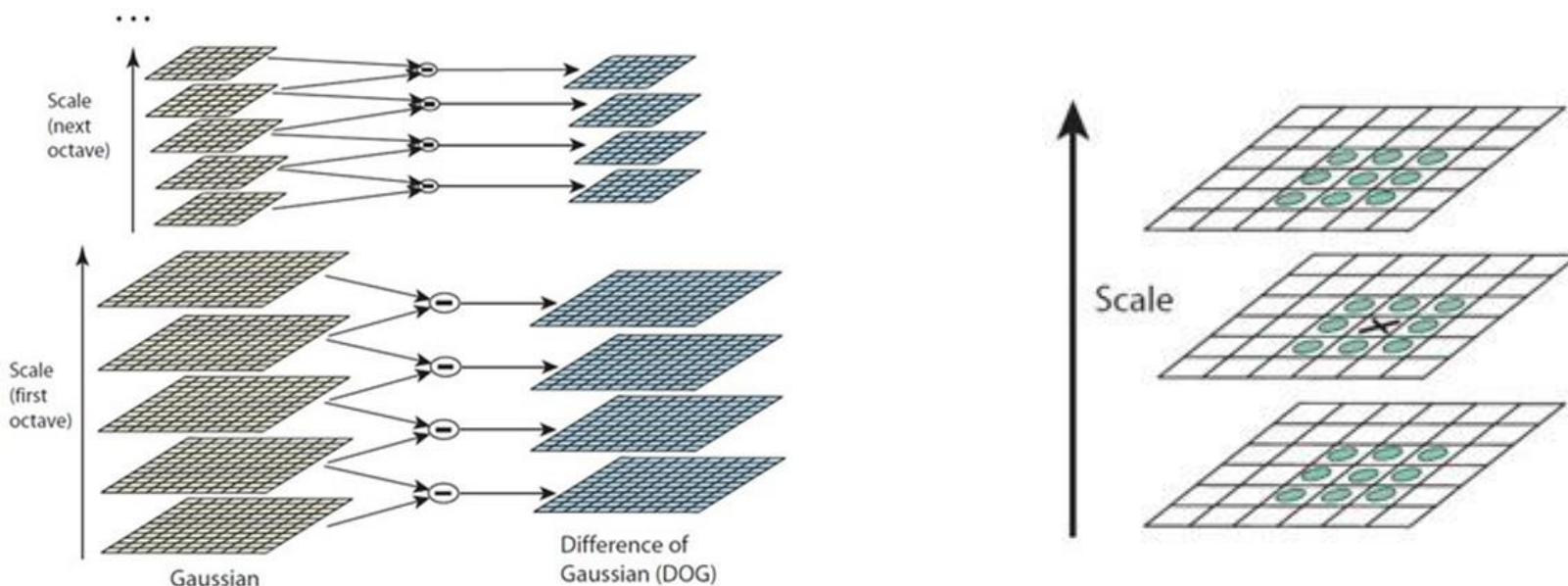
- 由于要在相邻尺度进行比较，如下图左侧每组含4层的高斯差分金字塔，只能在中间两层中进行两个尺度的极值点检测，其它尺度则只能在不同组中进行。
- 为了在每组中检测 S 个尺度的极值点，则DOG金字塔每组需 $S+2$ 层图像，而DOG金字塔由高斯金字塔相邻两层相减得到，则高斯金字塔每组需 $S+3$ 层图像，实际计算时 S 在3到5之间。
- 取 $S=3$ 时，假定高斯金字塔存储索引如下：

第0组(即第-1组): 0 1 2 3 4 5

第1组: 6 7 8 9 10 11

第2组: ?

则第2组第一张图片根据第一组中索引为9的图片降采样得到，其它类似。



构建尺度空间需确定的参数

- σ -尺度空间坐标
- O —组(octave)数
- S — 组内层数
- 在上述尺度空间中 σ , O 和 S 的关系如下:

$$\sigma(o, s) = \sigma_0 2^{o + \frac{s}{S}} \quad o \in [0, \dots, O-1], s \in [0, \dots, S+2]$$

- 其中 σ_0 是基准层尺度, o 为组(octave)的索引, s 为组内层的索引。关键点的尺度坐标就是按关键点所在的组和组内的层, 利用上述公式计算而来。

- 在最开始建立高斯金字塔时，要预先模糊输入图像来作为第0个组的第0层的图像，这时相当于丢弃了最高的空域的采样率。因此通常的做法是先将图像的尺度扩大一倍来生成第-1组。我们假定初始的输入图像为了抗击混淆现象，已经对其进行 $\sigma_{-1}=0.5$ 的高斯模糊，如果输入图像的尺寸用双线性插值扩大一倍，那么相当于 $\sigma_{-1}=1$ 。

- 取
$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y)$$

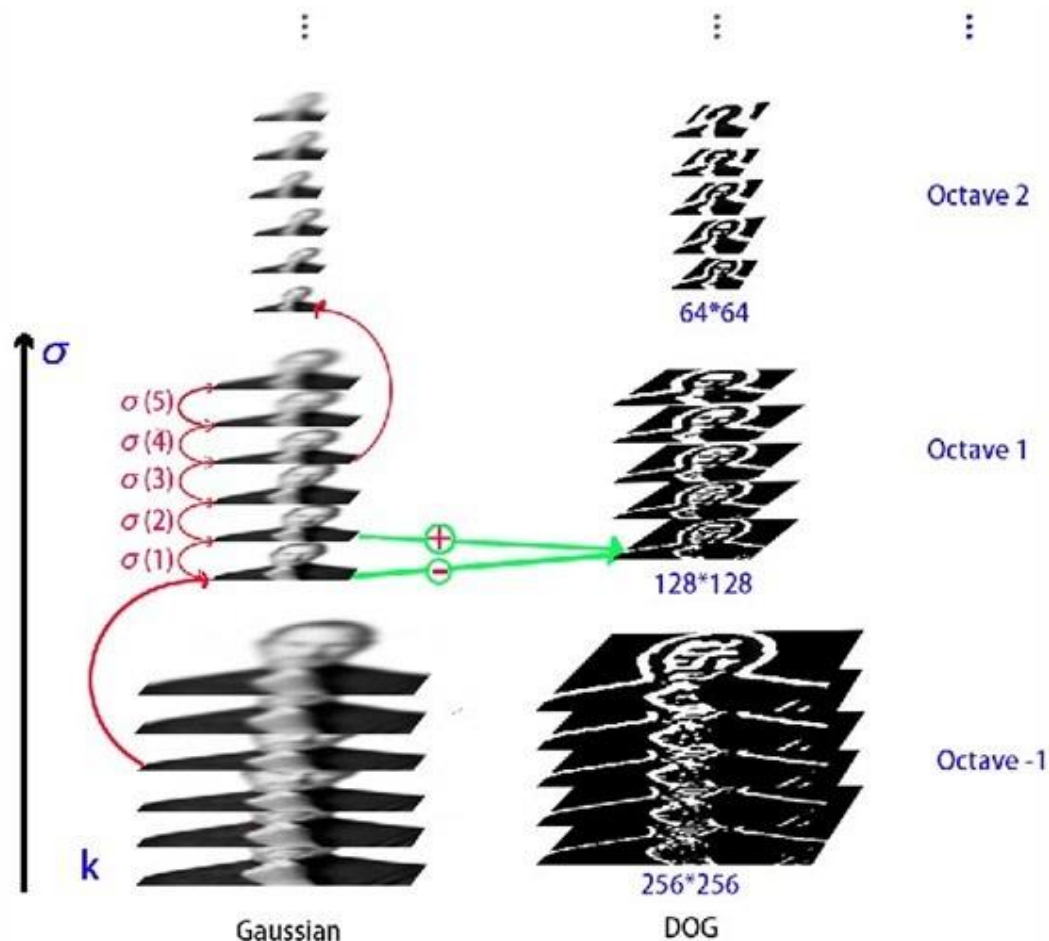
$$= L(x, y, k\sigma) - L(x, y, \sigma)$$
 式中的k为 $k = 2^{\frac{1}{s}}$
- 在构建高斯金字塔时，组内每层的尺度坐标按如下公式计算：

$$\sigma(s) = \sqrt{(k^s \sigma_0)^2 - (k^{s-1} \sigma_0)^2}$$

- 其中 σ_0 为初始尺度，lowe取 $\sigma_0=1.6$ ， $s=3$ ； s 为组内的层索引，不同组相同层的组内尺度坐标 $\sigma(s)$ 相同。组内下一层图像是由前一层图像按 $\sigma(s)$ 进行高斯模糊所得。上式用于一次生成组内不同尺度的高斯图像，而在计算组内某一层图像的尺度时，直接使用如下公式进行计算：

$$\sigma_{oct}(s) = \sigma_0 2^{\frac{s}{s}} \quad s \in [0, \dots, S+2]$$

- 式 $D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y)$ 可记作 $D(x, y, \sigma) = (G(x, y, \sigma(s+1)) - G(x, y, \sigma(s))) * I(x, y)$
 $= L(x, y, k\sigma) - L(x, y, \sigma)$ $= L(x, y, \sigma(s+1)) - L(x, y, \sigma(s))$
- 下图为构建DOG金字塔的示意图，原图采用128*128大小图像，扩大一倍后构建金字塔。

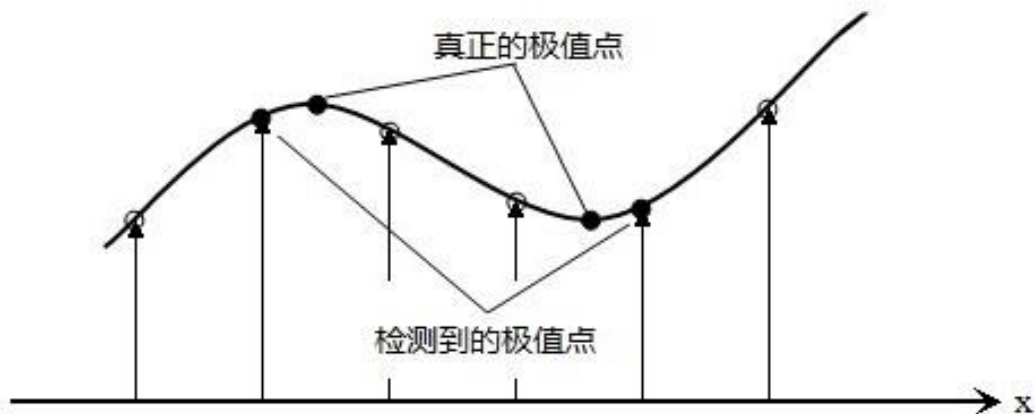




SIFT算法

- 尺度空间极值检测
- 关键点定位
- 关键点特征描述

- 之前的方法检测到的极值点是离散空间的极值点，下面通过拟合三维二次函数来精确确定关键点的位置和尺度，同时去除低对比度的关键点和不稳定的边缘响应点(因为DoG算子会产生较强的边缘响应)，以增强匹配稳定性、提高抗噪声能力。
- 离散空间的极值点并不是真正的极值点，下图显示了二维函数离散空间得到的极值点与连续空间极值点的差别。利用已知的离散空间点插值得到的连续空间极值点的方法叫做子像素插值 (Sub-pixel Interpolation) 。



- 候选特征点 x ，其偏移量定义为 Δx ，其对比度为 $D(x)$ 的绝对值 $|D(x)|$ ，对 $D(x)$ 在特征点 x 应用泰勒展开：

$$D(x) = D + \frac{\partial D^T}{\partial x} \Delta x + \frac{1}{2} \Delta x^T \frac{\partial^2 D}{\partial x^2} \Delta x$$

- 由于 x 是 $D(x)$ 的极值点，所以对上式求导并令其为0，得到极值点偏移量为

$$\Delta x = -\frac{\partial^2 D^{-1}}{\partial x^2} \frac{\partial D(x)}{\partial x}$$

- 对应极值点，方程的值为：
$$D(\hat{x}) = D + \frac{1}{2} \frac{\partial D^T}{\partial x} \hat{x}$$

- 其中 $\hat{x} = (x, y, \sigma)^T$ 代表相对插值中心的偏移量。当它在任一维度上的偏移量大于0.5时（即 x 或 y 或 σ ），意味着插值中心已经偏移到它的邻近点上，所以必须改变当前关键点的位置。同时在新的位置上反复插值直到收敛；也有可能超出所设定的迭代次数或者超出图像边界的范围，此时这样的点应该删除。另外， $|D(x)|$ 过小的点易受噪声的干扰而变得不稳定，所以将小于某个经验值(Lowe论文中使用0.03，Rob Hess等人实现时使用0.04/S)的极值点删除。同时，在此过程中获取特征点的精确位置(原位置加上拟合的偏移量)以及尺度($\sigma(o,s)$ 和 $\sigma_{oct}(s)$)。

消除边缘响应：

- 一个定义不好的高斯差分算子的极值在横跨边缘的地方有较大的主曲率，而在垂直边缘的方向有较小的主曲率。DOG算子会产生较强的边缘响应，需要剔除不稳定的边缘响应点。获取特征点处的Hessian矩阵，主曲率通过一个2x2 的Hessian矩阵H求出：

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

- H的特征值 α 和 β 代表x和y方向的梯度，

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta,$$

$$Det(H) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta$$

- Tr 表示矩阵H对角线元素之和， Det 表示矩阵H的行列式。假设是 α 较大的特征值，而是 β 较小的特征值，令 $\alpha=r\beta$ ，则

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\alpha + \beta)^2}{\alpha\beta} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r+1)^2}{r}$$

- D 的主曲率和 H 的特征值成正比，公式 $(r+1)^2/r$ 的值在两个特征值相等时最小，随着差异的增大而增大。值越大，说明两个特征值的比值越大，即在某一个方向的梯度值越大，而在另一个方向的梯度值越小，而边缘恰恰就是这种情况。所以为了剔除边缘响应点，需要让该比值小于一定的阈值，因此，为了检测主曲率是否在某域值 r 下，只需检测

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r},$$

上式成立时将关键点保留，反之剔除。取 $r = 10$ ，下图右侧为消除边缘响应后的关键点分布图。



消除前



消除后

有限差分法求导推导：

- 有限差分法以变量离散取值后对应的函数值来近似微分方程中独立变量的连续取值。在有限差分方法中，我们放弃了微分方程中独立变量可以取连续值的特征，而关注独立变量离散取值后对应的函数值。但是从原则上说，这种方法仍然可以达到任意满意的计算精度。因为方程的连续数值解可以通过减小独立变量离散取值的间隔，或者通过离散点上的函数值插值计算来近似得到。这种方法是随着计算机的诞生和应用而发展起来的。其计算格式和程序的设计都比较直观和简单，因而，它在计算数学中使用广泛。
- 有限差分法的具体操作分为两个部分：
 1. 用差分代替微分方程中的微分，将连续变化的变量离散化，从而得到差分方程组的数学形式；
 2. 求解差分方程组。
- 一个函数在 x 点上的一阶和二阶微商，可以近似地用它所临近的两点上的函数值的差分来表示。如对一个单变量函数 $f(x)$ ， x 为定义在区间 $[a,b]$ 上的连续变量，以步长 $h=\Delta x$ 将区间 $[a,b]$ 离散化，我们会得到一系列节点：
$$x_1 = a, x_2 = x_1 + h = a + h, \dots, x_{n+1} = x_n + h = b$$
- 然后求出 $f(x)$ 在这些点上的近似值。显然步长 h 越小，近似解的精度就越好。与节点 x_i 相邻的节点有 x_i+h 和 x_i-h ，所以在节点处可构造如下形式的差值： $f(x_i+h)-f(x_i)$ 节点的一阶向前差分； $f(x_i)-f(x_i-h)$ 节点的一阶向后差分； $f(x_i+h)-f(x_i-h)$ 节点的一阶中心差分。

- 使用中心差分法, 做如下推导:

函数 $f(x)$ 在 x_i 处的泰勒展开式为: $f(x) = f(x_i) + f'(x_i)(x - x_i) + \frac{f''(x_i)}{2!}(x - x_i)^2 + \dots + \frac{f^{(n)}(x_i)}{n!}(x - x_i)^n$,

则 $f(x_i - h) = f(x_i) + f'(x_i)((x_i - h) - x_i) + \frac{f''(x_i)}{2!}((x_i - h) - x_i)^2 + \dots$

$$f(x_i + h) = f(x_i) + f'(x_i)((x_i + h) - x_i) + \frac{f''(x_i)}{2!}((x_i + h) - x_i)^2 + \dots$$

忽略 h 平方之后的项, 联立上式, 解得: $f'(x_i) = \left(\frac{\partial f}{\partial x} \right)_{x_i} \approx \frac{f(x_i + h) - f(x_i - h)}{2h}$

$$f''(x_i) = \left(\frac{\partial^2 f}{\partial x^2} \right)_{x_i} \approx \frac{f(x_i + h) + f(x_i - h) - 2f(x_i)}{h^2}$$

二元函数的泰勒展开式如下:

$$\begin{aligned} f(x + \Delta x, y + \Delta y) &= f(x, y) + \Delta x \frac{\partial f(x, y)}{\partial x} + \Delta y \frac{\partial f(x, y)}{\partial y} \\ &+ \frac{1}{2!} \left[(\Delta x)^2 \frac{\partial^2 f(x, y)}{\partial x^2} + 2\Delta x \Delta y \frac{\partial^2 f(x, y)}{\partial x \partial y} + (\Delta y)^2 \frac{\partial^2 f(x, y)}{\partial y^2} \right] \\ &+ \frac{1}{3!} \left[(\Delta x)^3 \frac{\partial^3 f(x, y)}{\partial x^3} + 3(\Delta x)^2 \Delta y \frac{\partial^3 f(x, y)}{\partial x^2 \partial y} + 3\Delta x (\Delta y)^2 \frac{\partial^3 f(x, y)}{\partial x \partial y^2} + (\Delta y)^3 \frac{\partial^3 f(x, y)}{\partial y^3} \right] + \dots \end{aligned}$$

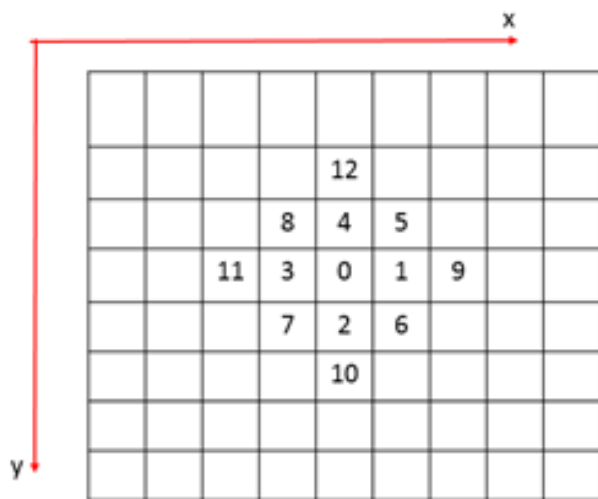
$$f(x_i + h, y_i + h), f(x_i + h, y_i + h), f(x_i + h, y_i + h) \text{ 和 } f(x_i + h, y_i + h)$$

- 将上式展开后忽略次要项联立解方程得二维混合偏导如下：

$$\begin{aligned} \frac{\partial^2 f(x_i, y_i)}{\partial x \partial y} \approx & \frac{1}{4h^2} [f(x_i + h, y_i + h) + f(x_i - h, y_i - h) \\ & - f(x_i + h, y_i - h) - f(x_i - h, y_i + h)] \end{aligned}$$

综上，推导了本小节遇到的所有导数计算。同理，利用多元泰勒展开式，可得任意偏导的近似差分表示。

- 在图像处理中，取 $h=1$ ，在左图所示的图像中，将像素0的基本中点导数公式整理如下：



$$\left(\frac{\partial f}{\partial x}\right)_0 = \frac{f_1 - f_3}{2h}$$

$$\left(\frac{\partial f}{\partial y}\right)_0 = \frac{f_2 - f_4}{2h}$$

$$\left(\frac{\partial^2 f}{\partial x^2}\right)_0 = \frac{f_1 + f_3 - 2f_0}{h^2}$$

$$\left(\frac{\partial^2 f}{\partial y^2}\right)_0 = \frac{f_2 + f_4 - 2f_0}{h^2}$$

$$\left(\frac{\partial^2 f}{\partial x \partial y}\right)_0 = \frac{(f_8 + f_6) - (f_5 + f_7)}{4h^2}$$

$$\left(\frac{\partial^4 f}{\partial x^4}\right)_0 = \frac{1}{h^4} [6f_0 - 4(f_1 + f_3) + (f_9 + f_{11})]$$

$$\left(\frac{\partial^4 f}{\partial y^4}\right)_0 = \frac{1}{h^4} [6f_0 - 4(f_2 + f_4) + (f_{10} + f_{12})]$$

$$\left(\frac{\partial^4 f}{\partial x^2 \partial y^2}\right)_0 = \frac{1}{h^4} [4f_0 - 2(f_1 + f_2 + f_3 + f_4) + (f_5 + f_6 + f_7 + f_8)]$$

- 若矩阵 $A = \begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{pmatrix}$ 可逆, 即

$$|A| = a_{00}a_{11}a_{22} + a_{01}a_{12}a_{20} + a_{02}a_{10}a_{21} - a_{00}a_{12}a_{21} - a_{01}a_{10}a_{22} - a_{02}a_{11}a_{20} \neq 0$$

- 则有

$$A^{-1} = \frac{1}{|A|} \begin{pmatrix} a_{11}a_{22} - a_{21}a_{12} & -(a_{01}a_{22} - a_{21}a_{02}) & a_{01}a_{12} - a_{02}a_{11} \\ a_{12}a_{20} - a_{22}a_{10} & -(a_{02}a_{20} - a_{22}a_{00}) & a_{02}a_{10} - a_{00}a_{12} \\ a_{10}a_{21} - a_{20}a_{11} & -(a_{00}a_{21} - a_{20}a_{01}) & a_{00}a_{11} - a_{01}a_{10} \end{pmatrix}$$



SIFT算法

- 尺度空间极值检测
- 关键点定位
- 关键点特征描述

邻域窗口梯度：

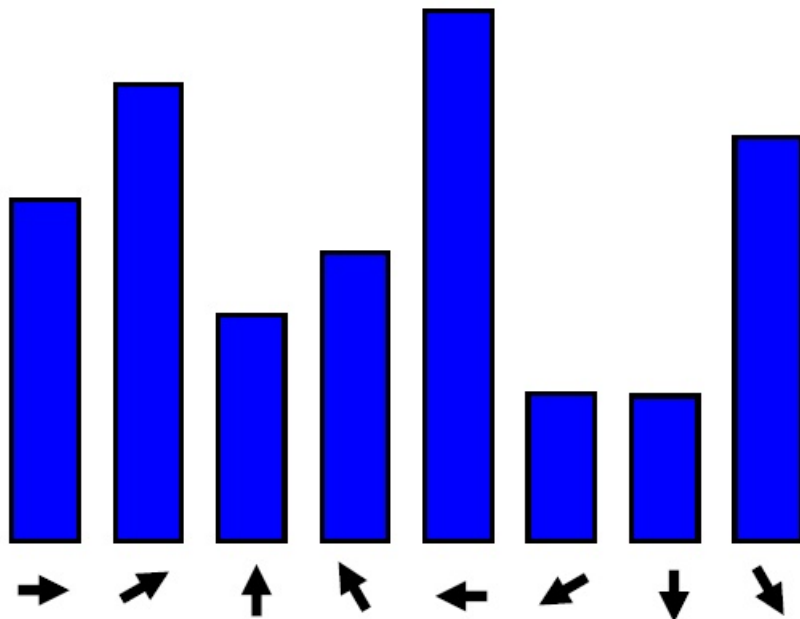
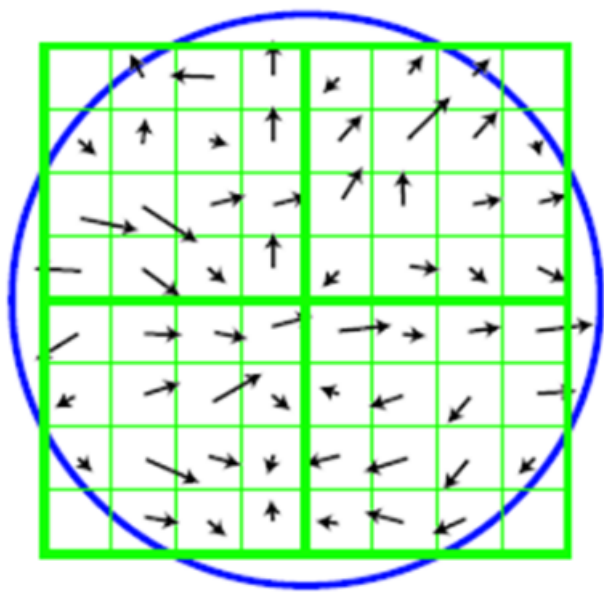
- 为了使描述符具有旋转不变性，需要利用图像的局部特征为给每一个关键点分配一个基准方向。使用图像梯度的方法求取局部结构的稳定方向。对于在DoG金字塔中检测出的关键点，采集其所在高斯金字塔图像 3σ 邻域窗口内像素的梯度和方向分布特征。梯度的模值和方向如下：

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$$
$$\theta(x, y) = \tan^{-1}((L(x, y+1) - L(x, y-1)) / (L(x+1, y) - L(x-1, y)))$$

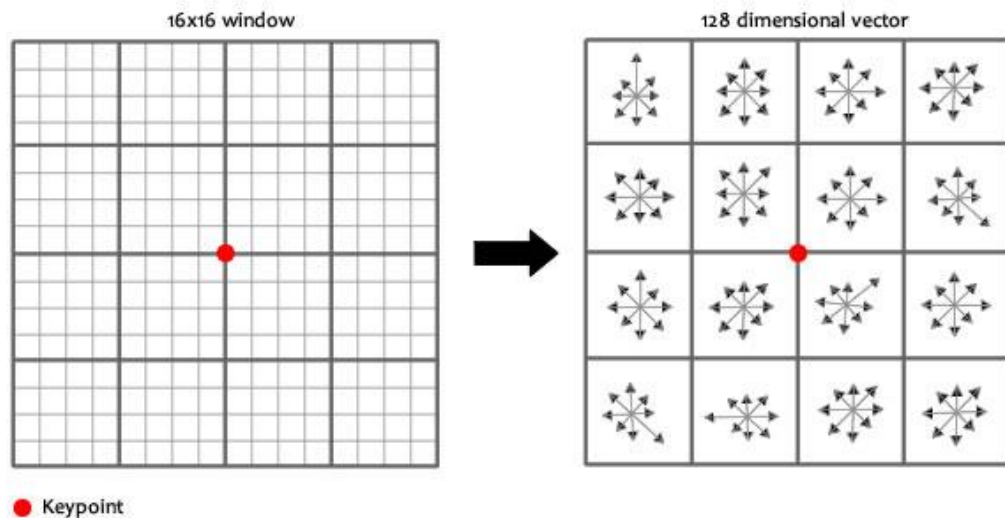
- L为关键点所在的尺度空间值，按Lowe的建议，梯度的模值 $m(x, y)$ 按 $\sigma = 1.5\sigma_{\text{oct}}$ 的高斯分布加成，按尺度采样的 3σ 原则，邻域窗口半径为 $3 \times 1.5\sigma_{\text{oct}}$ 。

方向直方图的生成:

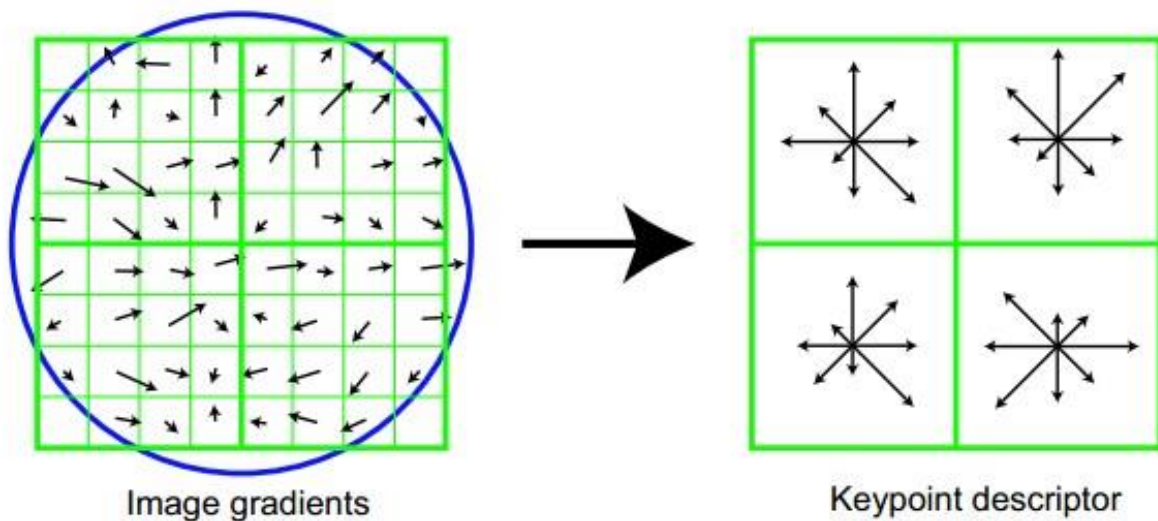
- 直方图统计半径为 $3 \times 1.5\sigma$ 。即计算以关键点为中心半径 $3 \times 1.5\sigma$ 邻域内所有点的梯度方向。
- 梯度直方图将 $0 \sim 360$ 度的方向范围分为36个柱。累计落到每个方向内的关键点个数加权值，以此生成梯度方向直方图。
- 将梯度方向直方图中纵坐标最大的项代表的方向分配给当前关键点作为主方向，若在梯度直方图中存在一个相当于主峰值80%能量的峰值，则将这个方向认为是关键点的辅方向。辅方向的设计可以增强匹配的鲁棒性，Lowe指出，大概有15%的关键点具有辅方向，而恰恰是这15%的关键点对稳定匹配起到关键作用。



- 对于每一个关键点，拥有三个信息：位置、尺度以及方向。接下来就是为每个关键点建立一个描述符，用一组向量将这个关键点描述出来，使其不随各种变化而改变，比如光照变化、视角变化等等。这个描述子不但包括关键点，也包含关键点周围对其有贡献的像素点，并且描述符应该具有较高的独特性，以便于提高特征点正确匹配的概率。
- SIFT描述子是关键点邻域高斯图像梯度统计结果的一种表示。通过对关键点周围图像区域分块，计算块内梯度直方图，生成具有独特性的向量，这个向量是该区域图像信息的一种抽象，具有唯一性。
- Lowe实验表明描述子采用在关键点尺度空间内 4×4 的窗口中计算的8个方向的梯度信息，共 $4 \times 4 \times 8 = 128$ 维向量表征，综合效果最优。



- 下图是一个SIFT描述子实例。其中描述子由 $2 \times 2 \times 8$ 维向量表征，也就是 2×2 个8方向直方图组成。左图的种子点由 8×8 单元组成。每一个小方格都代表了特征点邻域所在的尺度空间的一个像素，箭头方向代表了像素梯度方向，箭头长度代表了该像素的幅值。然后在 4×4 窗口内计算8个方向的梯度直方图，绘制每个梯度方向的累加可形成一个种子点。一个特征点由4个种子点的信息所组成。如图所示：



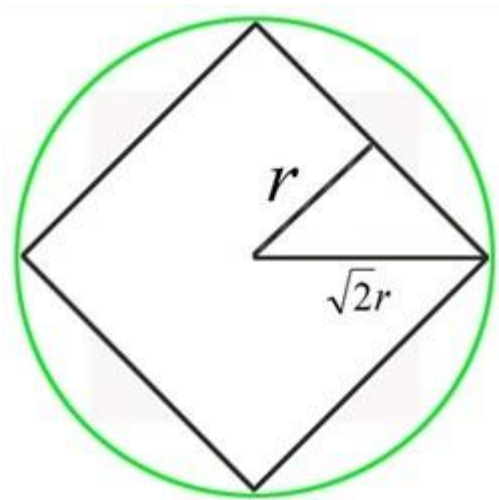
表示步骤如下：

- 1.确定计算描述子所需的图像区域

特征描述子与特征点所在的尺度有关，因此，对梯度的求取应在特征点对应的高斯图像上进行。

实际计算所需的图像区域半径为： $\text{radius} = (3\sigma_{\text{oct}} \times (d+1) \times \sqrt{2}) / 2$

计算结果四舍五入取整。（ σ_{oct} 为关键点所在组的组内尺度, $d=4$ ）

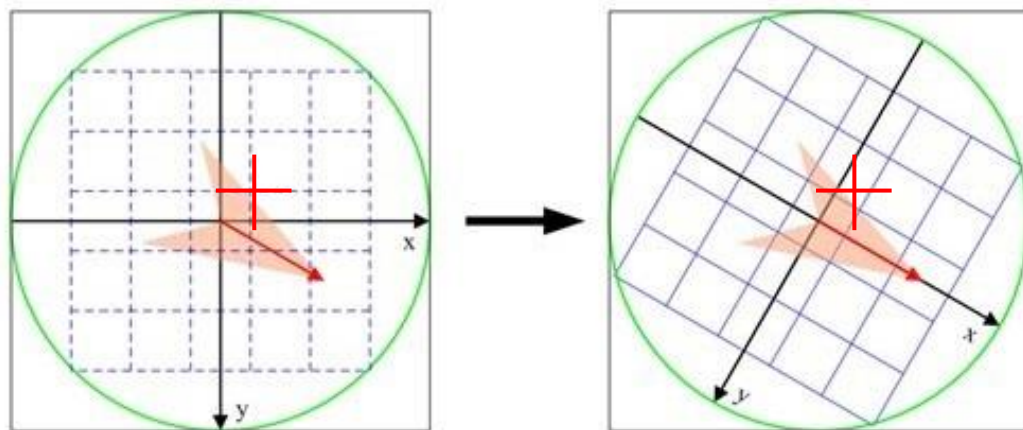


- 2.将坐标轴旋转为关键点方向，以确保旋转不变性

旋转后邻域内采样点的新坐标为：

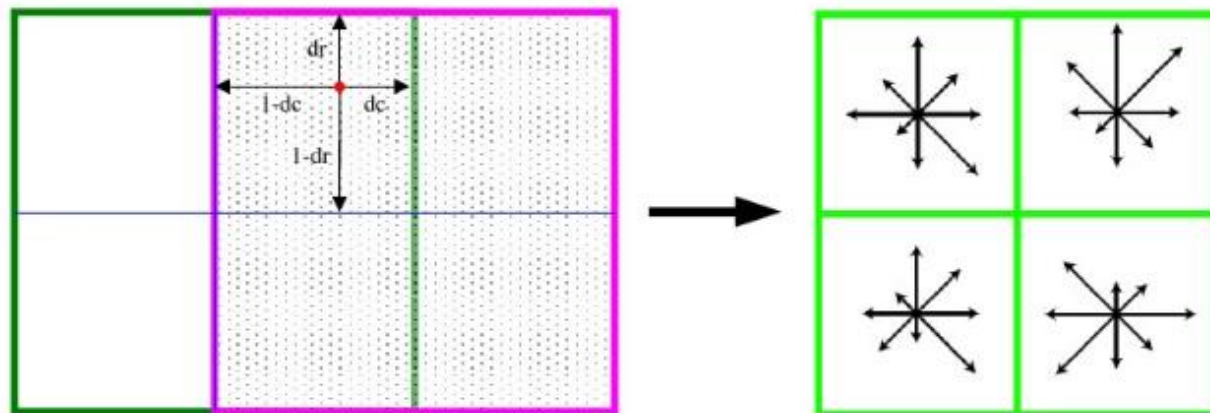
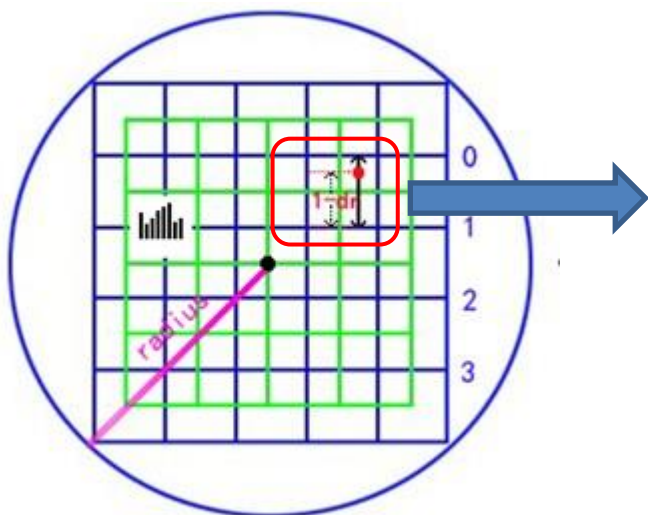
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

其中 $x, y \in [-\text{radius}, \text{radius}]$



坐标轴旋转

- 3.在图像半径区域内对每个像素点求其梯度幅值和方向，然后对每个梯度幅值乘以高斯权重参数，生成方向直方图。



$$weight = |grad(I_{\sigma}(x, y))| \times \exp\left(-\frac{x_k^2 + y_k^2}{2\sigma_w}\right) \times (1 - d_r) \times (1 - d_c) \times (1 - d_o)$$

x_k : 该点与关键点的列距离;

y_k : 该点与关键点的行距离;

σ_w : 等于描述子窗口宽度 3σ × 直方图列数 (取4) 的一半;

- 4.在窗口宽度为 2×2 的区域内计算8个方向的梯度直方图，绘制每个梯度方向的累加值，即可形成一个种子点。然后在下一个 2×2 的区域内进行直方图统计，生成下一个种子点，一共生成16个种子点。
- 5.描述子向量元素门限化以及门限化后的描述子向量规范化。
- 描述子向量元素门限化：
方向直方图每个方向上的梯度幅值限制在一定门限值之下（一般取0.2）

- 5. 描述子向量元素规范化:

描述子向量为 $H=(h_1, h_2, h_3 \dots h_{128})$,

归一化后的特征向量为 $L=(l_1, l_2, l_3 \dots l_{128})$, 则有
$$l_i = \frac{h_i}{\sqrt{\sum_{j=1}^{128} h_j^2}}, \quad j = 1, 2, 3, \dots$$

- 6. 描述子向量门限。非线性光照，相机饱和度变化对造成某些方向的梯度值过大，而对方向的影响微弱。因此设置门限值(向量归一化后，一般取0.2)截断较大的梯度值。
- 至此，SIFT特征描述向量生成。

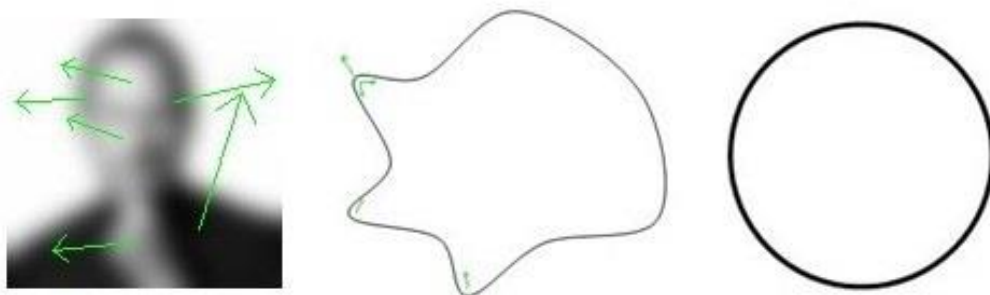
SIFT的缺点

SIFT在图像的不变特征提取方面拥有无与伦比的优势，但并不完美，仍然存在：

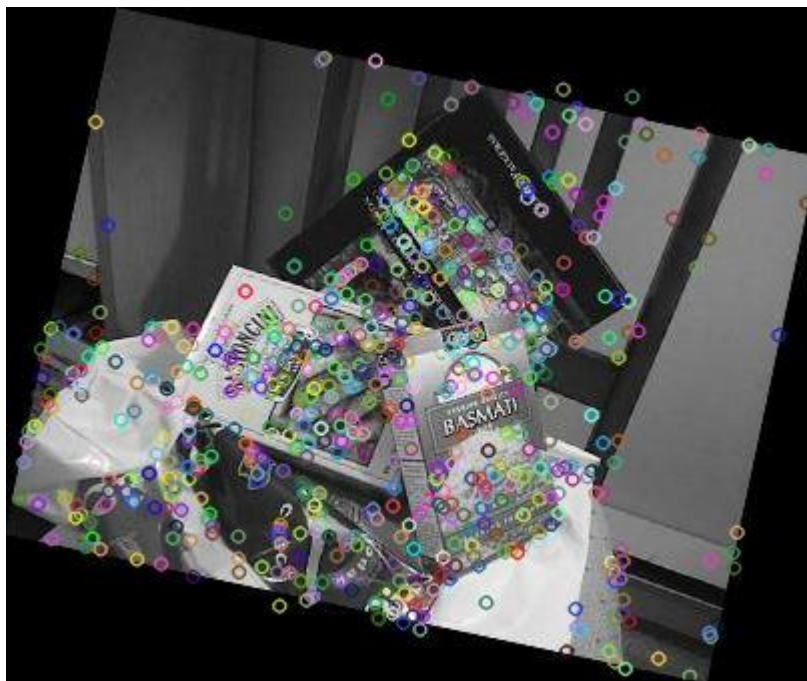
- 实时性差。
- 有时特征点较少。
- 对边缘光滑的目标无法准确提取特征点。

等缺点，如下图所示，对模糊的图像和边缘平滑的图像，检测出的特征点过少，对圆更是无能为力。

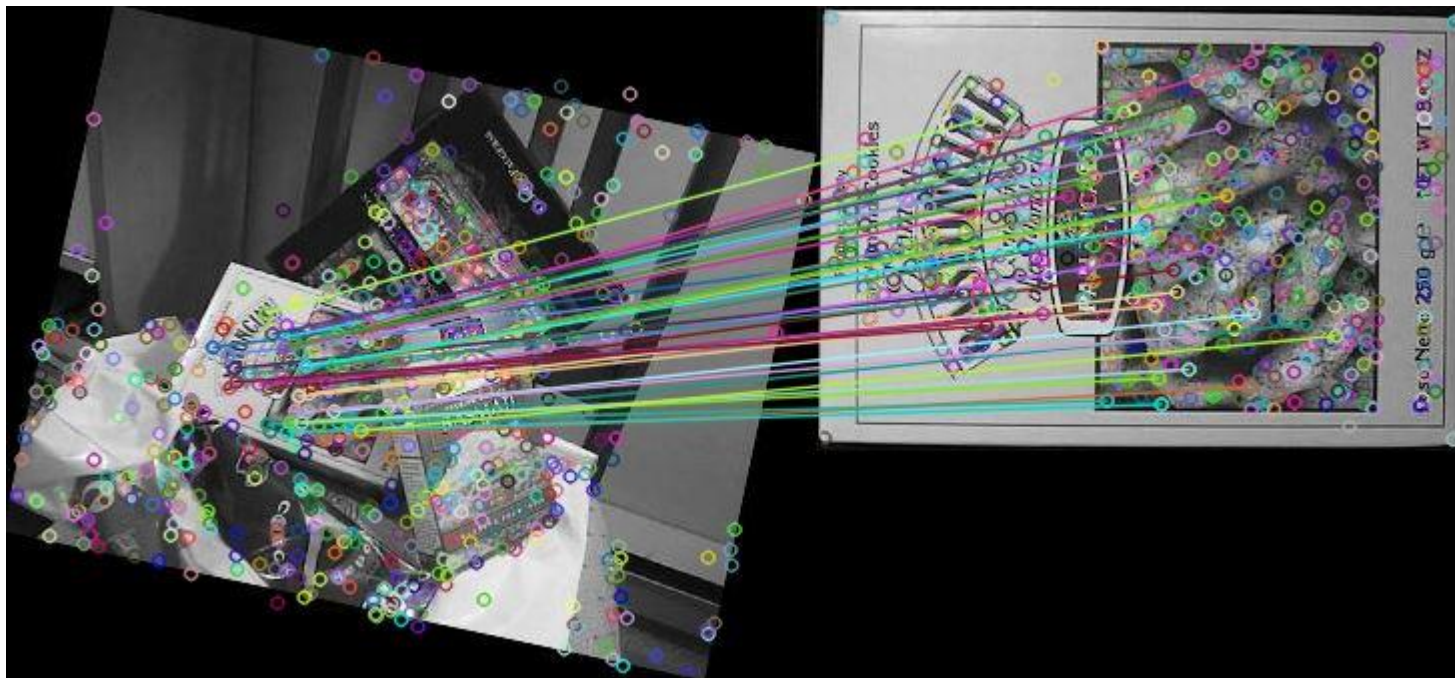
近来不断有人改进，其中最著名的有SURF；以及另辟蹊径的ORB等算子



结果示例：检测结果



结果示例：匹配结果



1. 尺度空间的核心思想是通过不同分辨率看同一个图像，可通过不同尺度的高斯函数与原始图像卷积实现
2. 极大值点的精确定位可通过解含有二阶导数项的方程得到
3. 借助在关键点附近区域的梯度方向统计与直方图生成，SIFT最终可得到对应每个关键点的128维向量描述。

EDU

CSDN学院 IT实战派

