

# 数字图像处理 SIFT & SURF 简述

胡欣毅 180776

信息科学与工程学院

icedomain\_hu@qq.com

2 组组长: 卞慧

胡欣毅	陈康	张可涵
包雅孟	周京鹏	郝培钧

2019 年 5 月 6 日



# 目录



# 目录



## 算法特点

- SIFT 特征是图像的局部特征，对旋转、尺度缩放、亮度变化保持不变性
- 独特性好，信息量丰富，适用于在海量特征数据库中进行快速、准确的匹配
- 多量性，即使少数的几个物体也可以产生大量的 SIFT 特征向量
- 高速性，经优化的 SIFT 匹配算法甚至可以达到实时的要求
- 可扩展性，可以很方便的与其他形式的特征向量进行联合



## 算法特点

- SIFT 特征是图像的局部特征，对旋转、尺度缩放、亮度变化保持不变性
- 独特性好，信息量丰富，适用于在海量特征数据库中进行快速、准确的匹配
- 多量性，即使少数的几个物体也可以产生大量的 SIFT 特征向量
- 高速性，经优化的 SIFT 匹配算法甚至可以达到实时的要求
- 可扩展性，可以很方便的与其他形式的特征向量进行联合

## 算法用途

- 目标的旋转、缩放、平移 (RST)
- 图像仿射/投影变换 (视点 viewpoint)
- 光照影响 (illumination)
- 目标遮挡 (occlusion)
- 杂物场景 (clutter)
- 噪声



## 尺度空间极值检测

- 高斯模糊
- 尺度空间极值检测
- 高斯金字塔的构建
- 空间极值点检测

# SIFT 算法



# SIFT 算法

## 尺度空间极值检测

- 高斯模糊
- 尺度空间极值检测
- 高斯金字塔的构建
- 空间极值点检测

## 关键点定位

- 关键点的精确定位
- 消除边缘响应
- ...



# SIFT 算法

## 尺度空间极值检测

- 高斯模糊
- 尺度空间极值检测
- 高斯金字塔的构建
- 空间极值点检测

## 方向确定

- 关键点方向分配
- ...

## 关键点定位

- 关键点的精确定位
- 消除边缘响应
- ...





# SIFT 算法

## 尺度空间极值检测

- 高斯模糊
- 尺度空间极值检测
- 高斯金字塔的构建
- 空间极值点检测

## 方向确定

- 关键点方向分配
- ...

## 关键点定位

- 关键点的精确定位
- 消除边缘响应
- ...

## 关键点描述

- 关键点特征描述
- ...



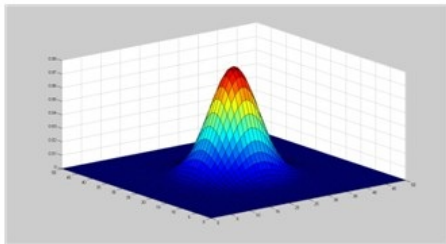
## 高斯模糊

高斯卷积核是实现尺度变换的唯一变换核，并且是唯一的线性核

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

$\sigma$  值越大，图像越模糊（平滑）

这样进行模糊处理比其它的均衡模糊滤波器更高地保留了边缘效果



## 图像的二维高斯模糊

因模板矩阵的关系而造成边缘图像缺失

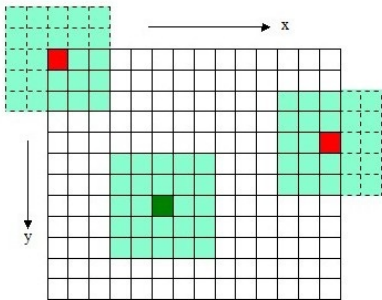


图:  $5 \times 5$  的高斯模板卷积



因模板矩阵的关系而造成边缘图像缺失

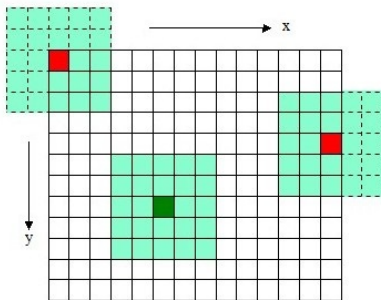


图:  $5 \times 5$  的高斯模板卷积

## 图像的二维高斯模糊

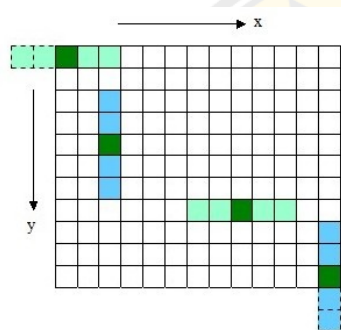


图:  $1 \times 5$  分离高斯模糊





## 高斯金字塔的构建

- 对图像做不同尺度的高斯模糊
- 对图像做降采样 (隔点采样)+ 高斯滤波
- 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张图像隔点采样得到

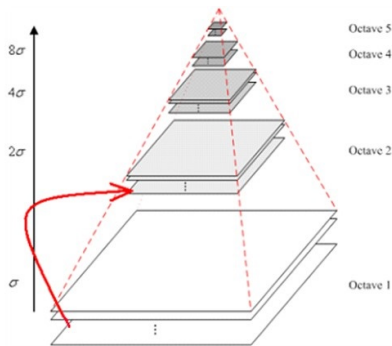


图: 高斯金字塔



- 对图像做不同尺度的高斯模糊
- 对图像做降采样 (隔点采样)+ 高斯滤波
- 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张图像隔点采样得到

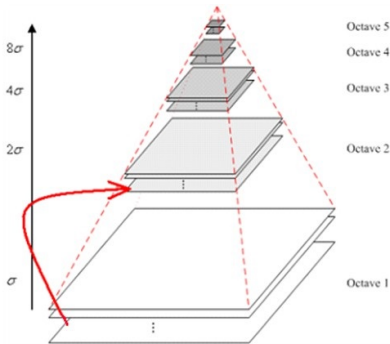


图: 高斯金字塔

## 高斯金字塔的构建

- 尺度规范化的 LoG 算子具有真正的尺度不变性
- Lowe 使用高斯差分金字塔近似 LoG 算子

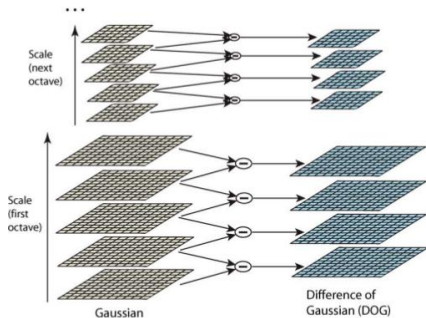


图: 高斯差分金字塔

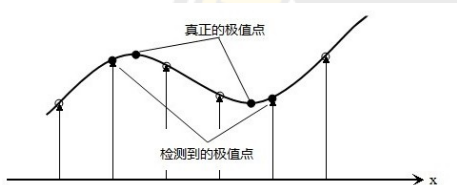
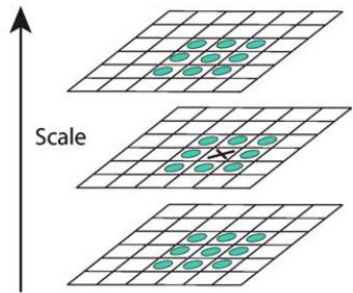


## 极值点检测 &amp; 关键点定位

同一组内各 DoG 相邻两层图像之间比较完成

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

子像素插值: 已知的离散空间点插值得到的连续空间极值点



需要消除“边缘响应”

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



## 关键点方向分配

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

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

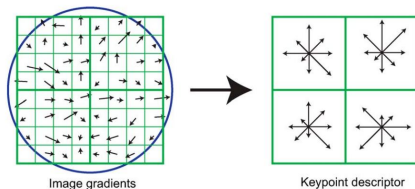


图: 关键点方向

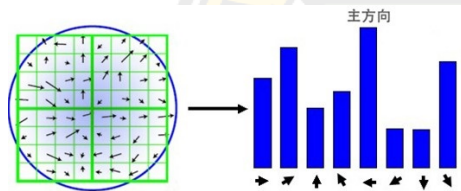


图: 关键点方向直方图



## 关键点方向分配

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

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

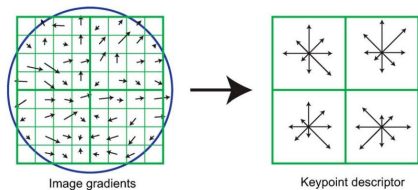


图: 关键点方向

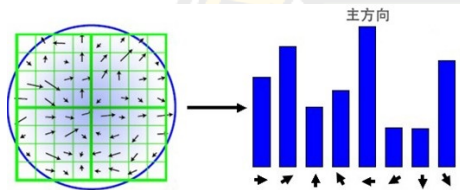


图: 关键点方向直方图

保留峰值大于主方向峰值 80% 的方向作为该关键点的辅方向

## 生成关键点特征描述

对于每一个关键点，拥有三个信息：位置、尺度以及方向。  
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。



## 生成关键点特征描述

对于每一个关键点，拥有三个信息：位置、尺度以及方向。  
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。

特征描述符生成的几个步骤：

- 确定计算描述子所需的图像区域



## 生成关键点特征描述

对于每一个关键点，拥有三个信息：位置、尺度以及方向。  
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。

特征描述符生成的几个步骤：

- 确定计算描述子所需的图像区域
- 校正旋转主方向，确保旋转不变性



## 生成关键点特征描述

对于每一个关键点，拥有三个信息：位置、尺度以及方向。  
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。

特征描述符生成的几个步骤：

- 确定计算描述子所需的图像区域
- 校正旋转主方向，确保旋转不变性
- 插值计算每个种子点八个方向的梯度



## 生成关键点特征描述

对于每一个关键点，拥有三个信息：位置、尺度以及方向。  
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。

特征描述符生成的几个步骤：

- 确定计算描述子所需的图像区域
- 校正旋转主方向，确保旋转不变性
- 插值计算每个种子点八个方向的梯度
- 生成描述子，最终形成一个 128 维的特征向量



## 生成关键点特征描述

对于每一个关键点，拥有三个信息：位置、尺度以及方向。  
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。

特征描述符生成的几个步骤：

- 确定计算描述子所需的图像区域
- 校正旋转主方向，确保旋转不变性
- 插值计算每个种子点八个方向的梯度
- 生成描述子，最终形成一个 128 维的特征向量
- 归一化处理，将特征向量长度进行归一化处理



## 生成关键点特征描述

对于每一个关键点，拥有三个信息：位置、尺度以及方向。  
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。

特征描述符生成的几个步骤：

- 确定计算描述子所需的图像区域
- 校正旋转主方向，确保旋转不变性
- 插值计算每个种子点八个方向的梯度
- 生成描述子，最终形成一个 128 维的特征向量
- 归一化处理，将特征向量长度进行归一化处理
- 描述子向量门限排除一定干扰





## 生成关键点特征描述

对于每一个关键点，拥有三个信息：位置、尺度以及方向。  
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。

特征描述符生成的几个步骤：

- 确定计算描述子所需的图像区域
- 校正旋转主方向，确保旋转不变性
- 插值计算每个种子点八个方向的梯度
- 生成描述子，最终形成一个 128 维的特征向量
- 归一化处理，将特征向量长度进行归一化处理
- 描述子向量门限排除一定干扰
- 按特征点的尺度对特征描述向量进行排序



校正旋转主方向

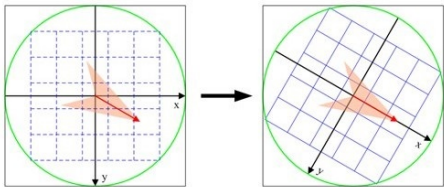


图: 坐标轴选择

## 生成关键点特征描述

生成描述子，最终形成一个 128 维的特征向量 ( $4 \times 4 \times 8$ )

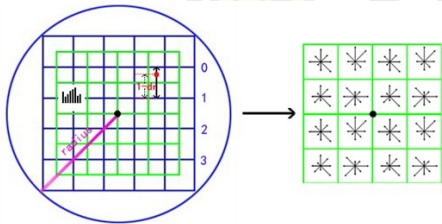


图: 描述子梯度

# 目录



## 构建 Hessian 阵

金字塔图像的获取

sift: DOG 图像

surf: Hessian 矩阵行列式近似值图像

每一个像素点都可以求出一个 Hessian 矩阵

$$H(f(x, y)) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}$$



## 构建 Hessian 阵

金字塔图像的获取

sift: DOG 图像

surf: Hessian 矩阵行列式近似值图像

每一个像素点都可以求出一个 Hessian 矩阵

$$H(f(x, y)) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}$$

当 Hessian 矩阵的判别式取得局部极大值 (结合下文特征点定位) 时, 判定当前点是比周围邻域内其他点更亮或更暗的点, 由此来定位关键点的位置。



## 构建尺度空间

不同组间图像的尺寸都是一致的

不同组间使用的盒式滤波器的模板尺寸逐渐增大

同一组间不同层间使用相同尺寸的滤波器，但是滤波器的模糊系数逐渐增大

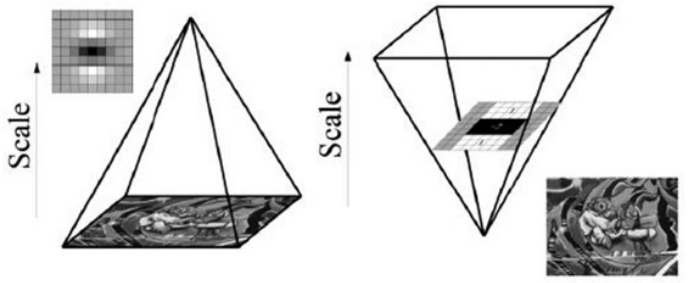
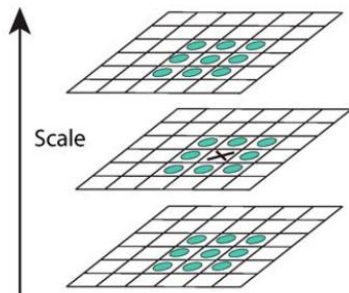


图: 盒式滤波器的模板尺寸



特征点的定位过程 Surf 和 Sift 保持一致，将经过 Hessian 矩阵处理的每个像素点与二维图像空间和尺度空间邻域内的 26 个点进行比较，初步定位出关键点，再经过滤除能量比较弱的关键点以及错误定位的关键点，筛选出最终的稳定的特征点。



图：特征点定位



## 特征点主方向分配

采用的是统计特征点圆形邻域内的 haar 小波特征。

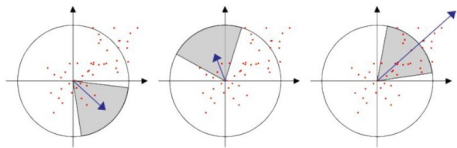


图: 特征点定位

在特征点的圆形邻域内，统计 60 度扇形内所有点的水平、垂直 haar 小波特征总和，然后扇形以一定间隔进行旋转并再次统计该区域内 haar 小波特征值之后，最后将值最大的那个扇形的方向作为该特征点的主方向。



- (沿着特征点的主方向) 在特征点周围取一个  $4 \times 4$  的矩形区域块
- 每个子区域统计 25 个像素的水平方向和垂直方向的 haar 小波特征 (水平和垂直方向相对主方向而言)
- 该 haar 小波特征为水平方向值之和、垂直方向值之和、水平方向绝对值之和以及垂直方向绝对值之和 ( $4 \times 4 \times 4$  维描述子)

## 生成特征点描述子

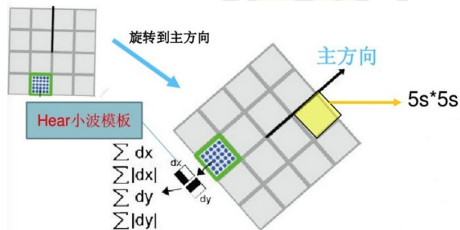


图: 特征点定位

# 特征点匹配

**同** Surf 也是通过计算两个特征点间的欧式距离来确定匹配度，欧氏距离越短，代表两个特征点的匹配度越好

**异** Surf 还加入了 Hessian 矩阵迹的判断

- 1 两个特征点的矩阵迹正负号相同，代表这两个特征具有相同方向上的对比度变化
- 2 两个特征点的矩阵迹正负号不同，说明这两个特征点的对比度变化方向是相反的



## ■ Sift

**优点** 特征稳定，对旋转、尺度变换、亮度保持不变性，对视角变换、噪声也有一定程度的稳定性

**缺点** 实时性不高，并且对于边缘光滑目标的特征点提取能力较弱



## ■ Sift

**优点** 特征稳定，对旋转、尺度变换、亮度保持不变性，对视角变换、噪声也有一定程度的稳定性

**缺点** 实时性不高，并且对于边缘光滑目标的特征点提取能力较弱

## ■ Surf

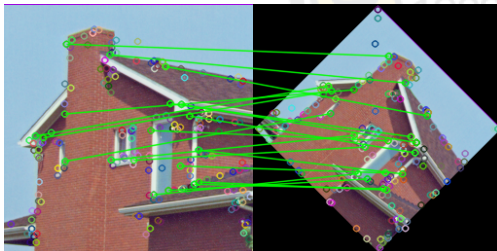
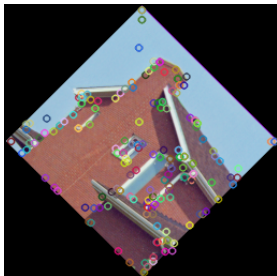
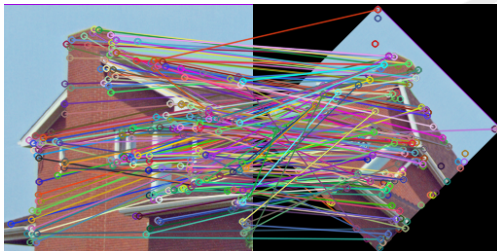
- 1 改进了特征的提取和描述方式
- 2 用一种更为高效的方式完成特征的提取和描述

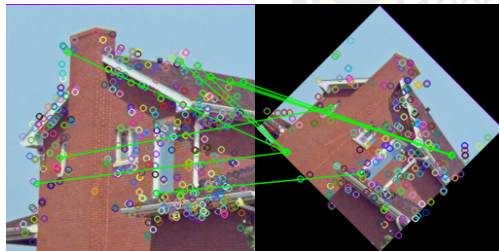
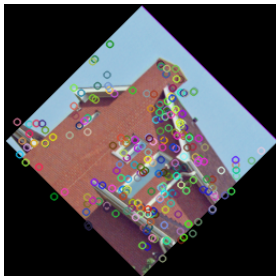
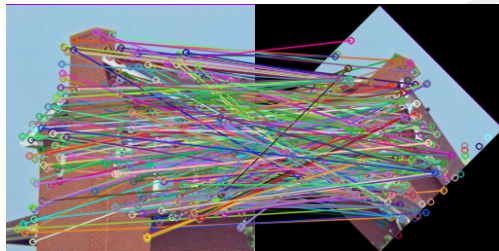


# 目录



# SIFT





# END-ING

