# 实验报告

## 实验目标

理解关键点检测算法DOG原理。

理解尺度变化不变特征SIFT。

采集一系列局部图像，自行设计拼接算法。

使用Python实现图像拼接算法。

### 具体要求

不允许使用现成的图像拼接程序。

在采图过程中可尽可能减少相机在垂直方向的运动，但不能假设图像只存在水平方向平移。

需包含图像融合部分，从而减少拼接图像中局部图像的“接缝”。

## 实现原理

### SHIFT特征提取算法

尺度不变特征变换(Scale-invariant feature transform， 简称SIFT)是图像局部特征提取的现代方法——基于区域/图像块的分析。其对旋转、尺度缩放、亮度变化保持不变性，对视角变化、仿射变换、噪声也保持一定程度的稳定性。

特征点是由DOG空间的局部极值点组成的。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1。

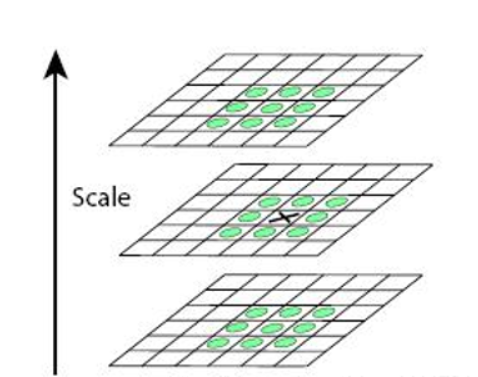
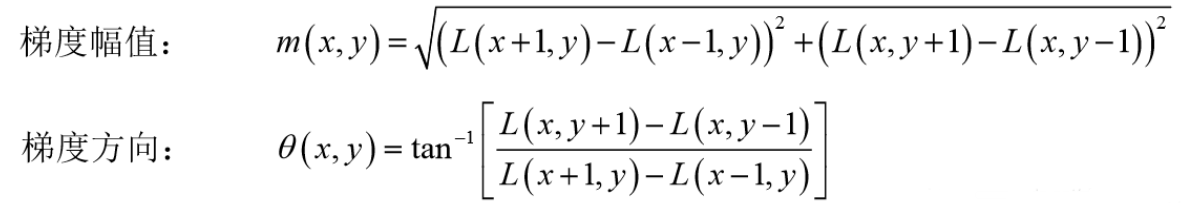
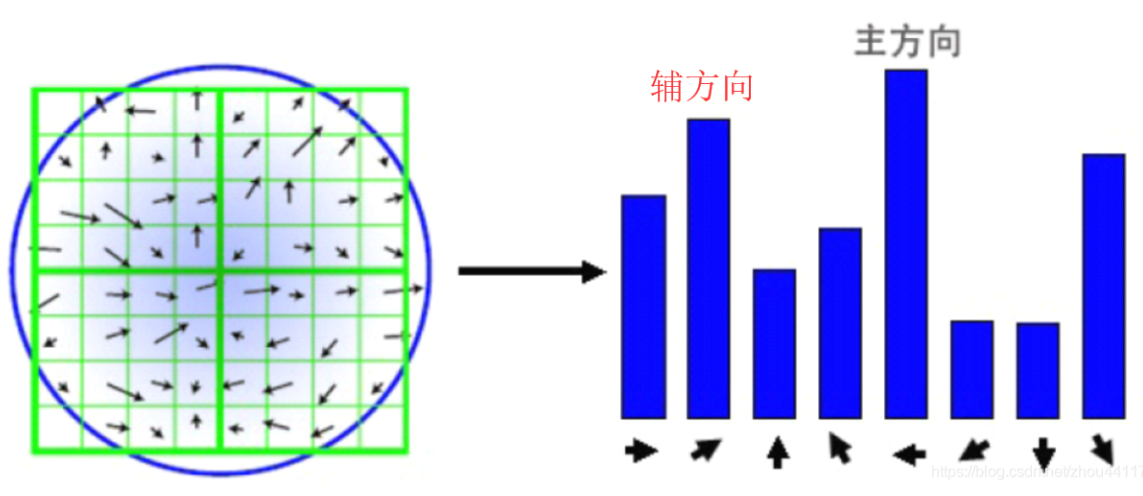


图1 中间的检测点和它同尺度的8个相邻点  
 经过上面的步骤已经找到了在不同尺度下都存在的特征点，为了实现图像旋转不变性，需要给特征点的方向进行赋值。利用特征点邻域像素的梯度来确定其方向参数，再利用图像的梯度直方图求取关键点局部结构的稳定方向。

我们通过每个极值点的梯度来为极值点赋予方向，梯度幅值等于上下两点像素值差的平方加上左右两点像素值差的平方，梯度方向则为上下两点像素值差与左右两点像素值差的商。

用特征点邻域像素的梯度来确定其方向参数，确定关键点的方向采用梯度直方图统计法，统计以关键点为原点，一定区域内的图像像素点对关键点方向生成所作的贡献。如图2。

图2 转换梯度直方图

通过以上步骤，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，使其不随各种变化而改变，比如光照变化、视角变化等等。并且描述符应该有较高的独特性，以便于提高特征点正确匹配的概率。  
 为了保证特征矢量具有旋转不变性，需要以特征点为中心，将特征点附近邻域内图像梯度的位置和方向旋转一个方向角θ，即将原图像x轴转到与主方向相同的方向。如图3。

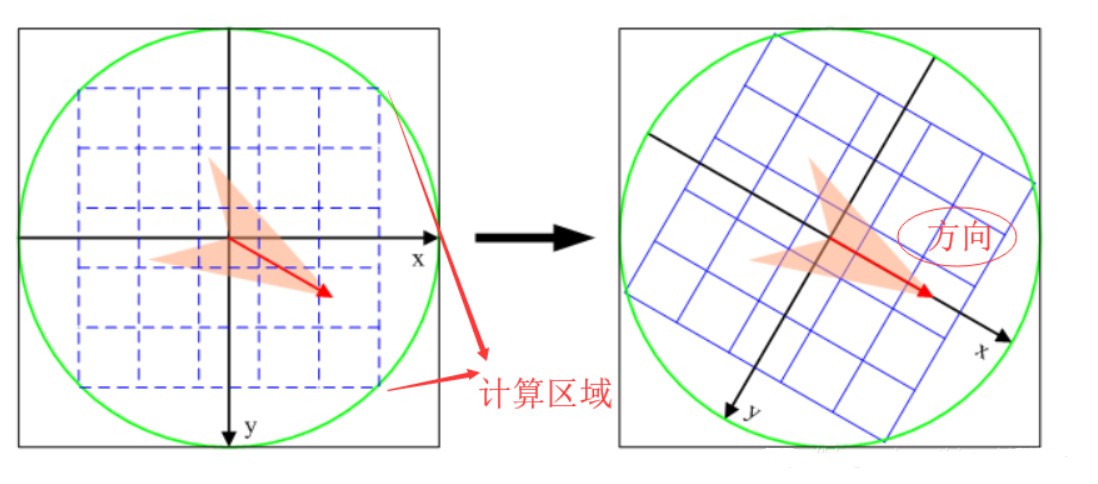
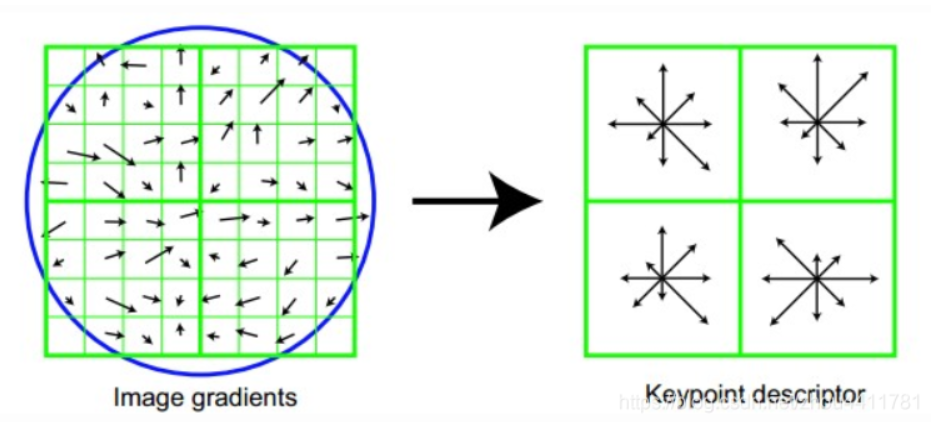
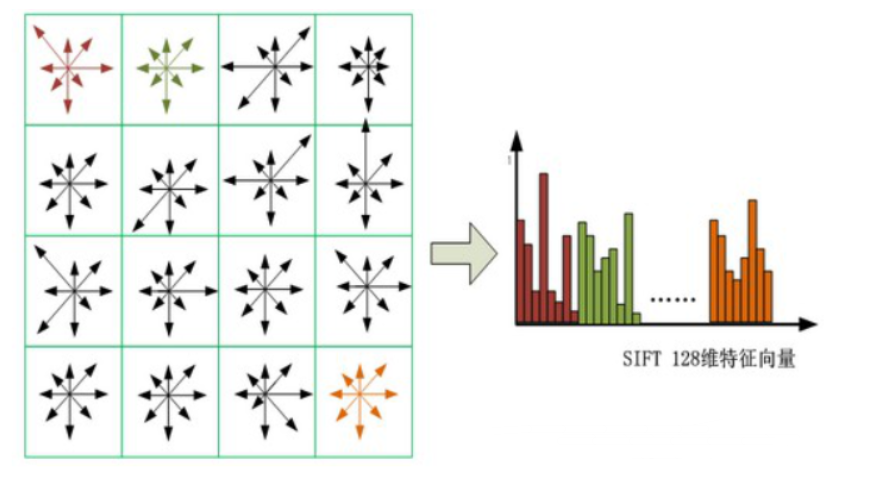
 图3图像特征梯度方向矫正

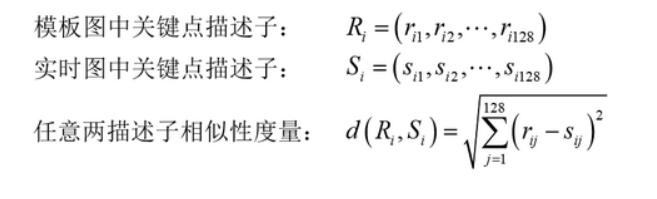
图4所示，左图的中央为当前关键点的位置，每个小格代表为关键点邻域所在尺度空间的一个像素，求取每个像素的梯度幅值与梯度方向，箭头方向代表该像素的梯度方向，长度代表梯度幅值，然后利用高斯窗口对其进行加权运算。最后在每个3σ×3σ的小块上(图像中每个正方形的区域的边长为4)绘制8个方向的梯度直方图，计算每个梯度方向的累加值，即可形成一个种子点，如右图所示。每个特征点由4个种子点组成，每个种子点有8个方向的向量信息。这种邻域方向性信息联合增强了算法的抗噪声能力，同时对于含有定位误差的特征匹配也提供了比较理性的容错性。

图4生成特征点描述子

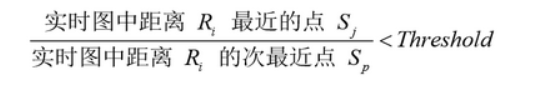
在实际的计算过程中，为了增强匹配的稳健性，Lowe建议对每个关键点使用4×4共16个种子点来描述，这样一个关键点就可以产生128维的SIFT特征向量。如图5。

  
图5 特征点描述向量的增广

对128维向量进行归一化处理，可以去除光照变化的影响。我们对模板图和目标图分别建立描述子集合。特征点的匹配是通过两点集合内关键点描述子的比对来完成，描述子的相似度量采用欧氏距离。假设如下：



设定阈值Threshold最终留下来的配对的关键点描述子，需要满足条件：



## 实验结果

输入A图像：



输入B图像：



运行结果

