
Image Stitching

Project Report

谢志旺 - 2015-05-14



Introduction

1.1 环境依赖和运行

依赖:

- gcc >= 4.7
- cmake
- gnu make
- opencv 2.4.9

运行:

- 运行时, 首先要确保图片在 image 文件夹中, 并且要保证图片出于竖直状态, 因为程序只能按顺序处理图片, 所以在程序里面硬编码了图片的输入, 也就是图片必须按从 0.jpg ~ n.jpg 的顺序。如果图片命名不是直接的数字, 可以直接修改程序 `main` 函数里面的图片读取部分。
- 在终端执行 `$bash run.sh` 来运行程序

结果:

- 运行结果在 result 文件夹, 名字为 result.jpg

Algorithm

2.1 特征检测

使用 SIFT 算子进行特征检测, SIFT 算法的实质是在不同的尺度空间上查找关键点(特征点), 并计算出关键点的方向。SIFT 所查找到的关键点是一些十分突出, 不会因光照, 仿射变换和噪音等因素而变化的点, 如角点、边缘点、暗区的亮点及亮区的暗点等。

2.1.1. 尺度空间极值检测

搜索所有尺度上的图像位置。通过高斯微分函数来识别潜在的对于尺度和旋转不变的兴趣点。

2.1.2 关键点定位

在每个候选的位置上, 通过一个拟合精细的模型来确定位置和尺度。关键点的选择依据于它们的稳定程度

2.1.3. 方向确定

基于图像局部的梯度方向, 分配给每个关键点位置一个或多个方向。所有后面的对图像数据的操作都相对于关键点的方向、尺度和位置进行变换, 从而提供对于这些变换的不变性。

2.1.4. 关键点描述

在每个关键点周围的邻域内，在选定的尺度上测量图像局部的梯度。这些梯度被转换成一种表示，这种表示允许比较大的局部形状的变形和光照变化。

2.1.5. 为关键点分配方向值

利用特征点领域像素的梯度方向分布特征来定关键点的方向，用梯度直方图来统计邻域像素的梯度方向。梯度方向的直方图的主峰值则代表了该关键点的主方向，如果有相当于主峰值的80%大小的其他峰值，则为该关键点的辅方向。可以看出关键点的方向就由一个主峰值方向和多个次峰值的方向决定。这样可以减少图像旋转对特征关键点的影响。

2.1.6. 生成特征向量描述子

为了进一步描述关键点的信息，则确定关键点的邻域范围的大小很重要。如 Fig1 所示：每个小方格代表的是关键的邻域像素，小方格内的的箭头代表了该邻域像素的梯度方向，箭头的大小代表了梯度大小。Fig1(b)中的左上角的小方块由 Fig1(a)中的左上角的四个小方格组成。即：(b) 图中的每个小方块的方向是对于 (a) 图中的方格方向的累积值。Fig1 显示的是 (a) 图是 8×8 的邻域范围，(b) 图显示的是 2×2 的种子点（每个方格代表一个种子）。为了增强抗噪能力和匹配的稳健性，通常把邻域的取值范围设成 16×16 ，那么就会产生 4×4 的种子点。这样每个关键点的信息量就包含在了 $4\times 4\times 8=128$ 维特征向量里。

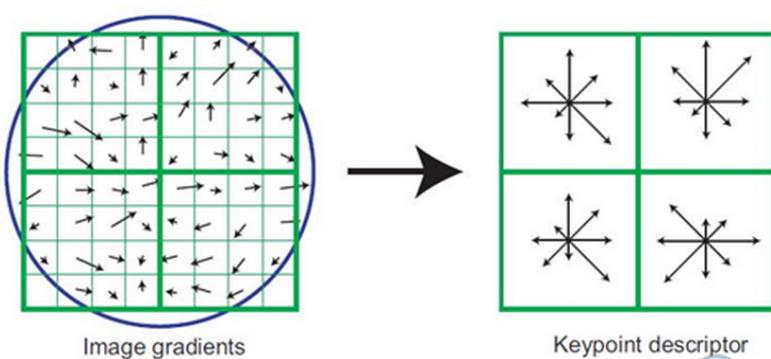


Fig 1

2.1.7. 特征向量的匹配

对SIFT特征向量进行匹配是根据相似性度量来进行的，常用的匹配方法有：欧式距离和马氏距离等。采用欧氏距离对SIFT的特征向量进行匹配。获取SIFT特征向量后，采用优先 k-d 树进行优先搜索来查找每个特征点的近似最近邻特征点。在这两个特征点中，如果最近的距离除以次近的距离少于某个比例阈值，则接受这一对匹配点。降低这个比例阈值，SIFT匹配点数目会减少，但更加稳定。匹配的结果如 Fig 2所示：

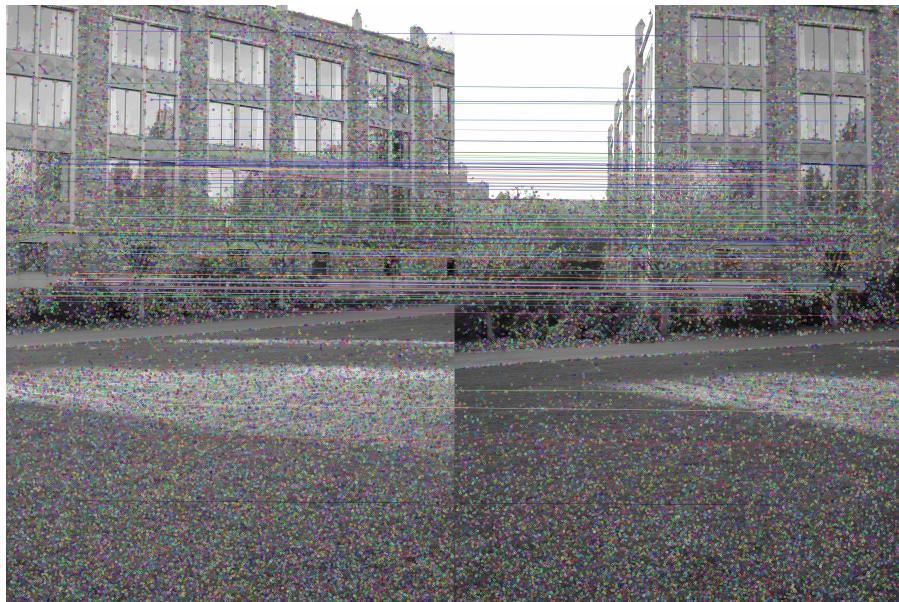


Fig 2

2.2 变形

图像的变形包括平面投影，球面投影，柱面投影和立方体投影，这里采用柱面投影。柱面投影的核心是以相机为圆柱中心点，相机焦距为半径的一个柱面作为投影面。相比立方体和球面投影方式，柱面投影只能在水平方向上转动，在垂直方向上的转动有一定的限制，只允许在较小的范围上转动。投影之前效果如 Fig 3，之后的效果如 Fig4 所示：



Fig 3





Fig 4

2.3 图像变换

通常待拼接的图像之间一般都有不同程度的几何形变，为了实现图像的拼接，就必须找到图像之间存在的几何变换模型，空间变换模型的方式有平移模型、刚性模型、相似模型、仿射模型以及投影模型等，仿射变换和透视变换如下：

1) 仿射变换

仿射变换的特点是：变换是线性的，经过仿射变换的直线仍然是直线。可以表示相机的平移、旋转、缩放运动。仿射变换后的平行线依然是平行线，仿射变换有6个自由度。矩阵为：

$$\text{Homography I: } H = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

2) 透视变换

透视变换是最一般的变换形式，前面提到的几种变换：刚性变换、仿射变换等都是透视变换的特殊形式，透视变换具有8个自由度，如下所示（其中 a_{33} 为1）：

$$\text{Affine: } A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

2.4 变形校正

有了前面的操作，我们就可以得到一个弯曲的图像，该图像是卷曲，如 Fig5，这时我们需要将图像拉直

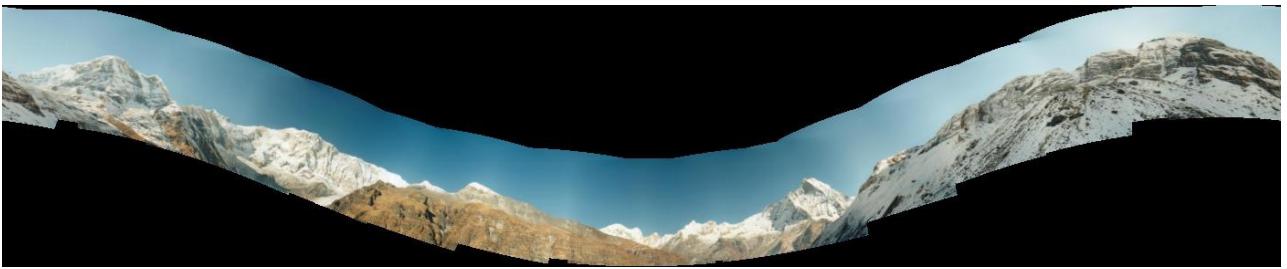


Fig 5

首先在第一次搜索进行时，我们首先将头部和尾部连接，然后执行循环，确保头部和尾部对齐。这样就可以得到一个更长的对齐的图片。结果如 Fig6 所示。



Fig 6

2.5 融合和裁剪

图像配准后，通过对配准图像重采样就可以进行图像的拼接。但由于图像灰度差异等原因，拼接后的图像很容易出现亮度差异和拼接接缝，所以在图像拼接后需要进行图像融合，使拼接后的图片看起来自然准确。融合后，可能出现一些黑边，这个时候利用裁剪算法将黑边去掉并进行映射。结果如 Fig7 所示



Fig 7

Reference

Matthew Brown and David G. Lowe Automatic Panoramic Image Stitching using Invariant Features. Department of Computer Science, University of British Columbia, Vancouver, Canada.