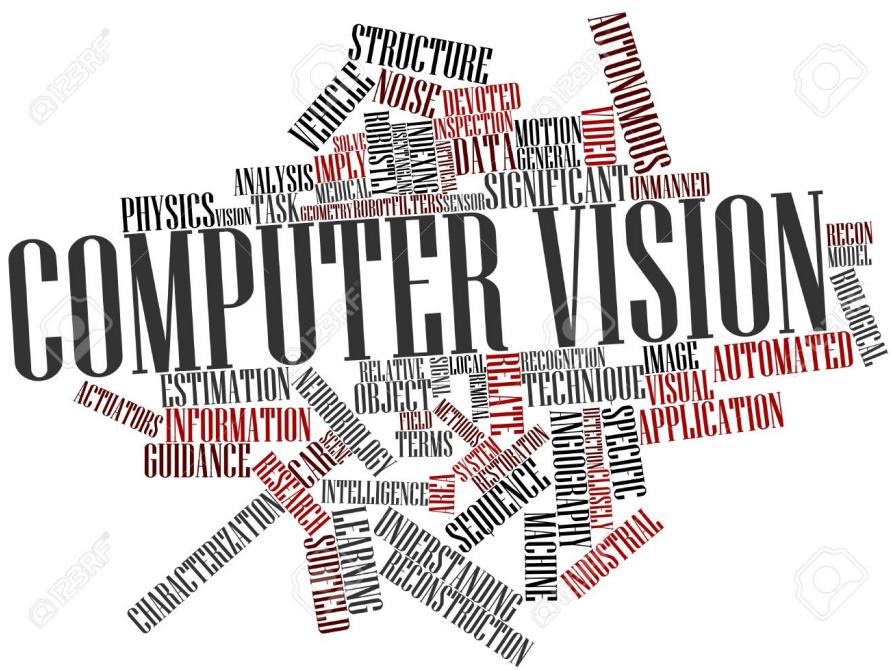


# Computer Vision HW1



安捷 1601210097

2017 年 4 月 10 日

## 1 算法实现介绍

在这一次作业中，我按照作业要求，分别实现了针对线性模型的 RANSAC 算法和针对线性最小二乘问题的 RANSAC 算法；我使用 vl-feat 工具包与我自己实现的 RANSAC 算法编写了图像拼接的函数和程序，并实现了任意多张图像的拼接；在作业中我还尝试了使用新老照片共同进行图像拼接的实验，没有取得比较好的效果。

在这里有几点算法实现的细节需要说明：

1. 为了使得图像最终的拼接效果反差不至于太大，起到一定的融合效果，我在算法中使用了两个操作，首先，对输入图像进行灰度均值的平均化，其次，对拼接的图像进行中值滤波以去除接缝处的灰度差异，考虑到最终对拼接图像的观察效果，我注释掉了中值滤波的代码；
2. 除了最终显示拼接结果的代码外，其余显示图像中间状态及匹配效果的图像显示代码我都进行了注释，这是因为 cat 操作在多图拼接的过程中会遇到图像大小不一致的异常，两幅图像拼接过程中，可以自行取消注释，即可看到拼接的中间过程；
3. 只需按照 three\_image\_mosaic.m 中的样式即可实现任意多张图像的拼接，但是当图像拼接达到一定规模之后，RANSAC 算法的参数将难以适应，我尝试了拼接四张图像，在最后一张图像的拼接过程中始终没有得到比较好的结果，三张图像的拼接较为稳定；

## 2 脚本参数设置

参数名称	参数值	参数含义
N	4	求解线性最小二乘所需的最小点数
K	300	RANSAC 算法迭代次数
T	2	判断样本适合模型的误差阈值
D	60	判断模型是否合适的样本数阈值
PEAK_THRESH	0.5	SIFT 算法阈值
EDGE_THRESH	10	SIFT 算法阈值

表 1: two\_image\_mosaic.m 参数表

## 3 运行结果

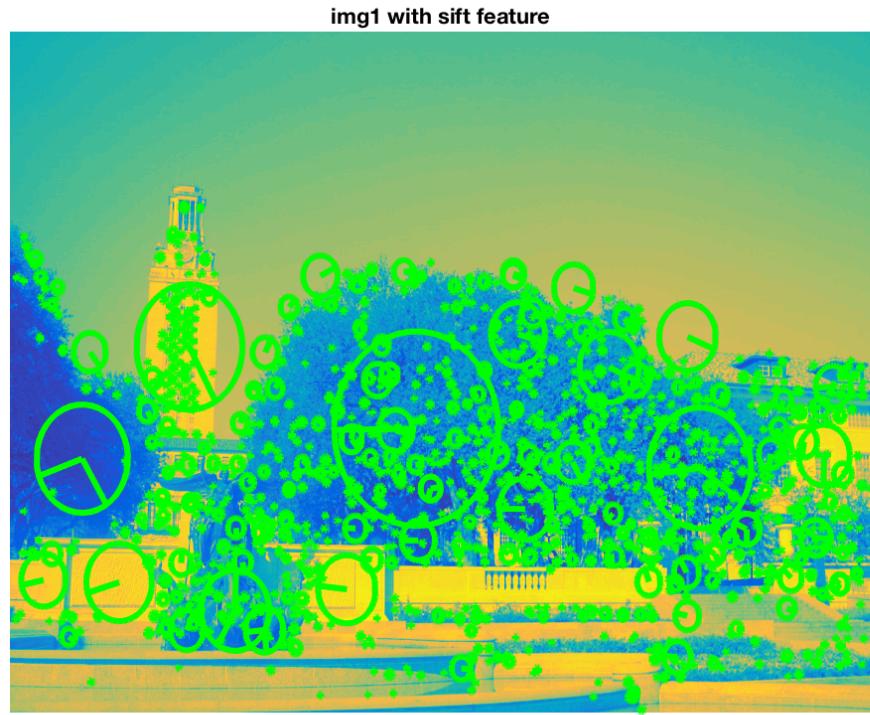


图 1: 左图像 SIFT 检测子结果

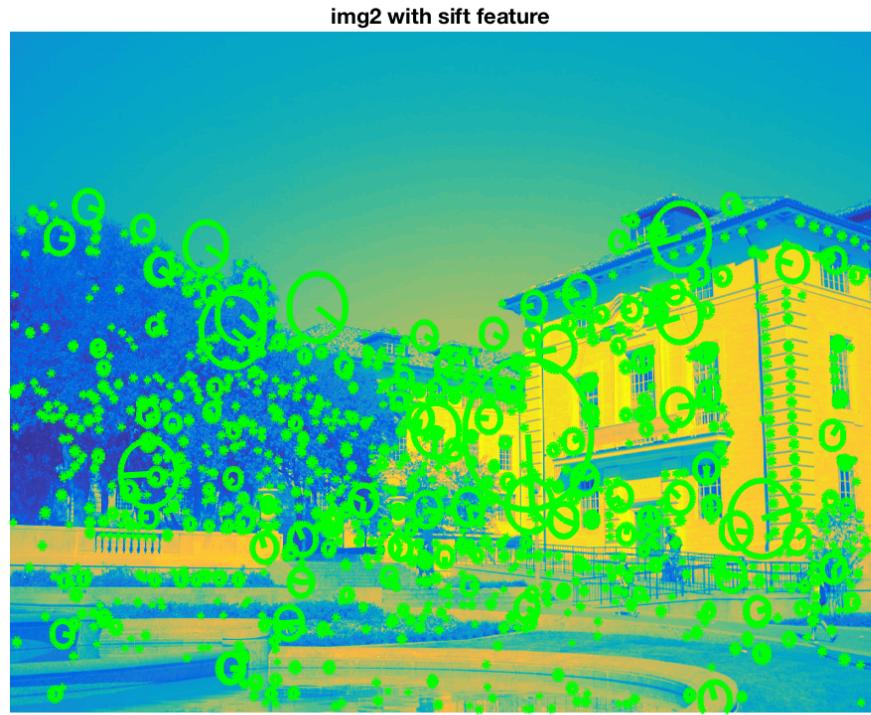


图 2: 右图像 SIFT 检测子结果

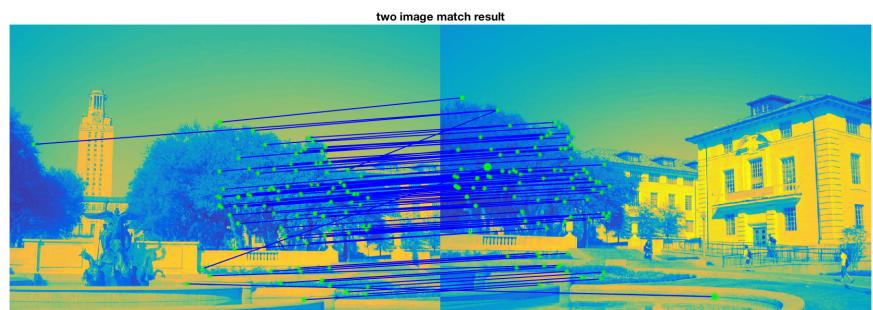


图 3: SIFT 匹配结果

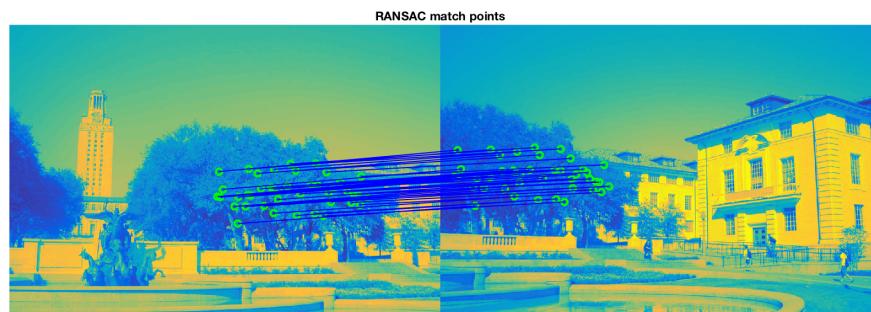


图 4: RANSAC 匹配结果

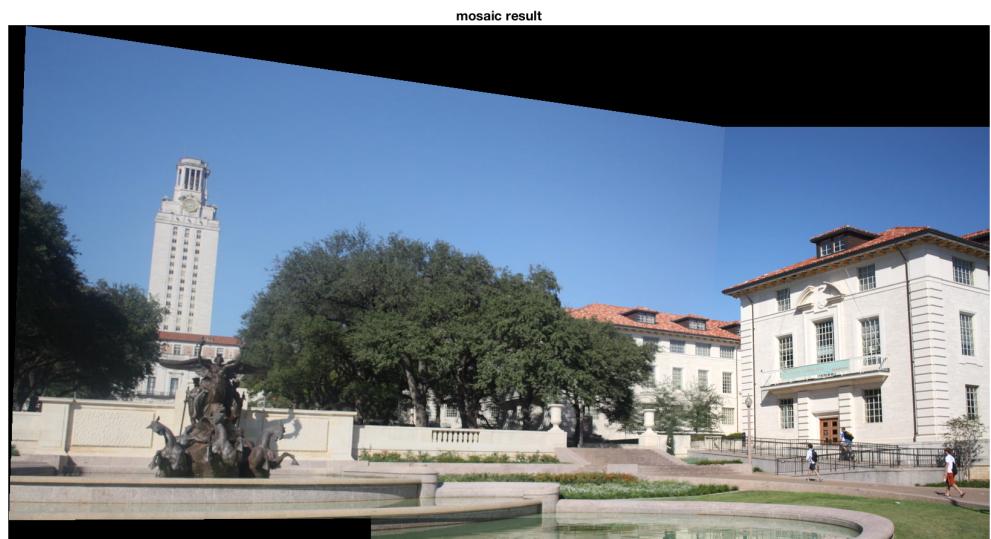


图 5: 两图像拼接结果

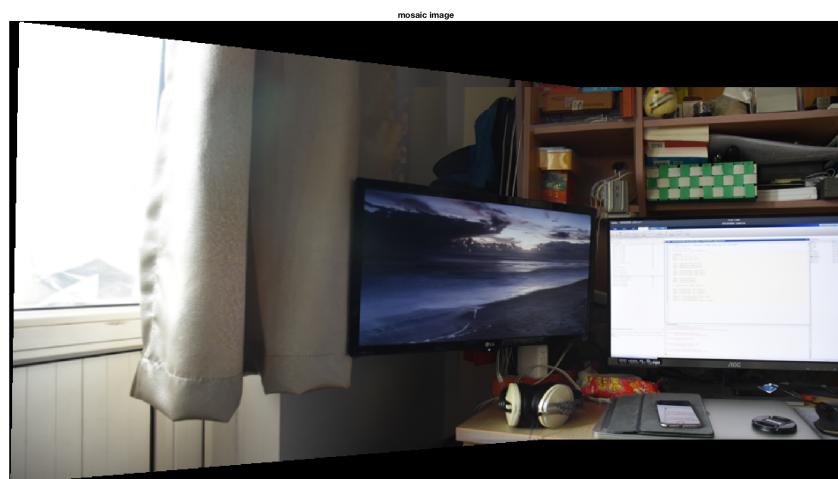


图 6: 三图像拼接结果 1

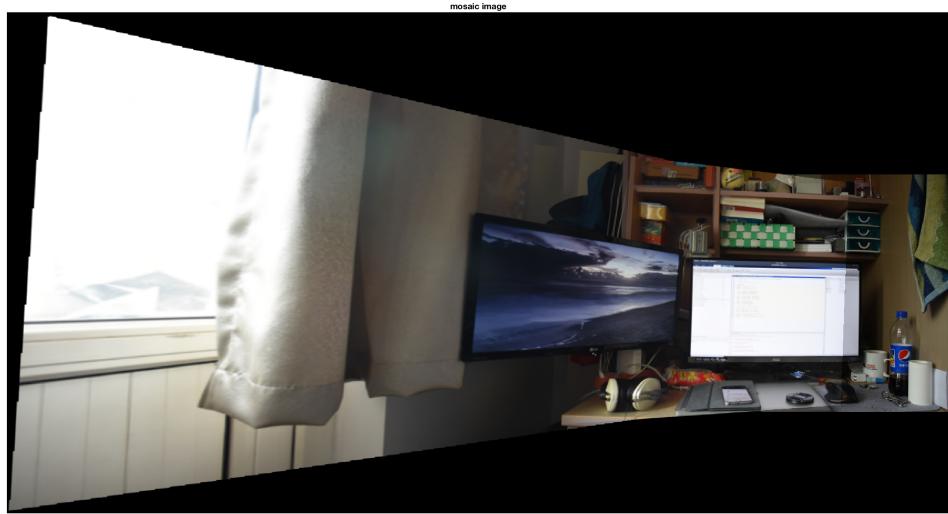


图 7: 三图像拼接结果 2

## 4 结论

1. SIFT 算子与 RANSAC 算法结合可以实现较为精确的图像拼接；
2. RANSAC 算法对于参数非常敏感，尤其是误差阈值 T 和模型最小支持样本阈值 D，我在实现的过程中尝试了多次调整参数才在三张图像的拼接中得到一个可以接受的结果，我观察到，对于两张图像的拼接，参数的敏感性稍微可以减弱；
3. 上述算法能够更好的工作且具有较高的鲁棒性，但是对于图像的尺度差别过大的情况下，由于 SIFT 算子不再可以很好的表示两者尺度差异巨大的特征，算法不 work，这部分的数值实验可以参看 two\_image\_mosaic\_test.m 中新旧图像拼接的部分；

## 5 软件版本及测试平台信息

这部分内容请参看源代码所在文件夹内的 README 文件。