

综合实训期末总结

学号：15331406 姓名：张音

此次实训中，本人主要完成图像局部特征匹配的工作。基于老师提供的论文和图像集，采用如下方法进行图像间的局部特征匹配及标注。

一、实现方法

1. 对图像集手动分类

进行实验的图像集较大，且每个景点所包含的建筑数量繁多，再加上拍摄照片时的客观因素（如光照、清晰度等），因此在实验前对图像集手动分类，主要将建筑物或建筑物的一部分进行归类，例如“岭南_陈家祠_屋檐雕饰”一类，即包含陈家祠景区某屋檐的整体、局部或多角度图像。

2. 利用 SIFT 方法抽取特征并匹配

SIFT 是用于描述具有尺度不变性，可在图像中检测出关键点的一种局部特征描述子。使用 Matlab 工具，利用基于尺度不变性的 SIFT 方法对分类的图像集中每张图进行特征提取：

```
%将图像数据转换为single类型
im1 = im2single(im1);
im2 = im2single(im2);

%转换成灰度图像
if size(im1,3)>1, im1g=rgb2gray(im1); else im1g=im1; end
if size(im2,3)>1, im2g=rgb2gray(im2); else im2g=im2; end

%SIFT matches
%提取特征点（特征点坐标，描述子）
[f1,d1] = vl_sift(im1g);
[f2,d2] = vl_sift(im2g);
```

其中 f 存储特征点的坐标，尺度和方向，d 存储描述子（维度为 128*特征点个数）

3. 对两幅图的特征进行匹配

此步骤主要完成匹配特征点的工作，使用 vl_ubcmatch 函数，返回匹配点索引和匹配度：

```
%根据特征点匹配，默认阈值为1.5
[matches, scores] = vl_ubcmatch(d1,d2);
%获取匹配点个数
numMatches = size(matches,2);

%获取两幅图的匹配点及对应描述子
X1 = f1(1:2,matches(1,:)); X1(3,:) = 1;
X2 = f2(1:2,matches(2,:)); X2(3,:) = 1;
```

4. 对匹配特征进行筛选

上一步获取的匹配特征点索引，仅仅起到初筛匹配点的作用，对于保留下来的特征点，还应进行再次筛选，这里采用 RANSAC 和单应性变换的方法，从匹配点集中随机抽选 RANSAC 样本，也即 4 个匹配点对，在对其进行变换矩阵 H 的计算。根据得出的变换矩阵、样本集和误差度量函数计算满足当前变换矩阵的一致集，并返回一致集中元素个数，若当前一致集中元素个数最多，则设为最优一致集，更新错误概率直至小于设定阈值，具体实现如下：

```
%迭代100次，选择最佳H
for t=1:100
    %抽选匹配点对
    subset = vl_colsubset(1:numMatches, 4); %随机抽取四列
    A = [];
    for i=subset
        %求两个矩阵的Kronecker积, 就是矩阵a中的每个元素都乘以矩阵b。
        A = cat(1, A, kron(X1(:, i)', vl_hat(X2(:, i)))); %vl_hat斜对称矩阵
    end
    %奇异值分解
    [U, S, V] = svd(A);
    %得到变换矩阵
    H{t} = reshape(V(:, 9), 3, 3) ;

    %经变换矩阵计算对应点
    X2_ = H{t} * X1;
    du = X2_(1, :)./X2_(3, :) - X2(1, :)./X2(3, :);
    dv = X2_(2, :)./X2_(3, :) - X2(2, :)./X2(3, :);
    %若误差小于20*20, 则保留特征点
    ok{t} = (du.*du + dv.*dv) < 20*20;
    %记录保留特征点个数
    score(t) = sum(ok{t});
end

[score, best] = max(score);
%选取一致集元素个数最大的对应的H变换矩阵
H = H{best};
ok = ok{best};

function err = residual(H)
    u = H(1)*X1(1, ok)+H(4)*X1(2, ok)+H(7);
    v = H(2)*X1(1, ok)+H(5)*X1(2, ok)+H(8);
    d = H(3)*X1(1, ok)+H(6)*X1(2, ok)+1;
    du = X2(1, ok)- u./d;
    dv = X2(2, ok)-v./d;
    err = sum(du.*du+dv.*dv);
end
```

5. 计算筛选特征点的比例

对于保留下来的特征点，统计其个数并计算相对于初筛特征点的比例。将分类后图像集中的图片两两匹配(注意 `sift_mosaic(im1, im2)`和 `sift_mosaic(im2, im1)`得到的结果并不相同)), 并将其配对比例输出至 `match.mat` 文件中。

6. 确定主建筑图像

由于一组图像中，涉及到建筑的多角度和多部位，需要确定一个主图像，以便在主图像上进行匹配部分的标注。

具体方法是读取存储的匹配比例文件 `match.mat`，对图像 `i` 参与匹配的所有匹配点百分比相加，再取均值，计算每一张图片的匹配均值，取该值最大的图像作为主建筑图像。

```
mat_path = dir(strcat(sub_file_path, '*.*.mat'));
mat_name = mat_path(1).name;
mat = load(strcat(sub_file_path, mat_name));
match_matrix=mat.match_rate;
[row, col]=size(match_matrix);
sum_row=sum(match_matrix, 2)';
average_percent=(sum_row-100)./(row-1);
[result, index]=max(average_percent);
```

7. 存储匹配局部的边界点

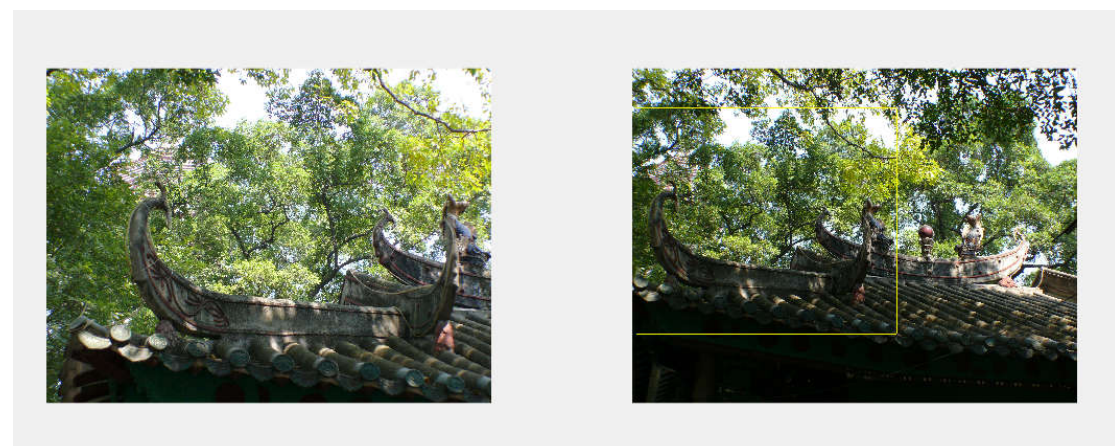
选出主建筑图像后，再次调用 `sift_mosaic` 函数，将匹配区域 `x` 和 `y` 坐标的最大和最小值保存至 `xmin`, `xmax`, `ymin`, `ymax` 中，并存至 `match_position.mat` 文件：

```
xmin=min(f1(1, matches(1, ok)));
xmax=max(f1(1, matches(1, ok)));
ymin=min(f1(2, matches(1, ok)));
ymax=max(f1(2, matches(1, ok)));
```

8. 标注匹配区域

根据主建筑索引和匹配边界点索引，注明主建筑图像与其他图像间匹配的区域。

实验结果如下：



二、实现效果

1. 岭南_陈家祠\屋顶

待匹配图像:

图 1



图 2

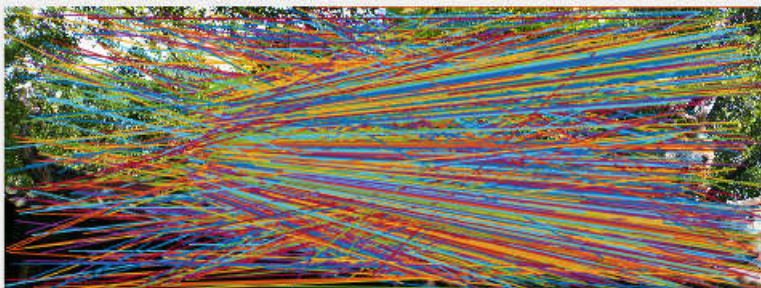


图 3



筛选匹配点:

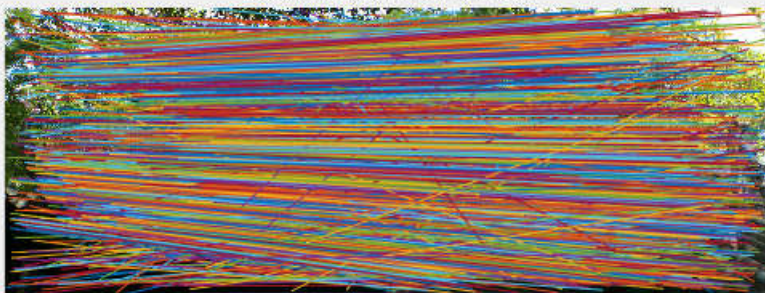
1553 tentative matches



611 (39.34%) inliner matches out of 1553



1936 tentative matches



1288 (66.53%) inliner matches out of 1936



判断主建筑图像为图 1，存储其相对于其他图像的匹配信息，并标注匹配信息：



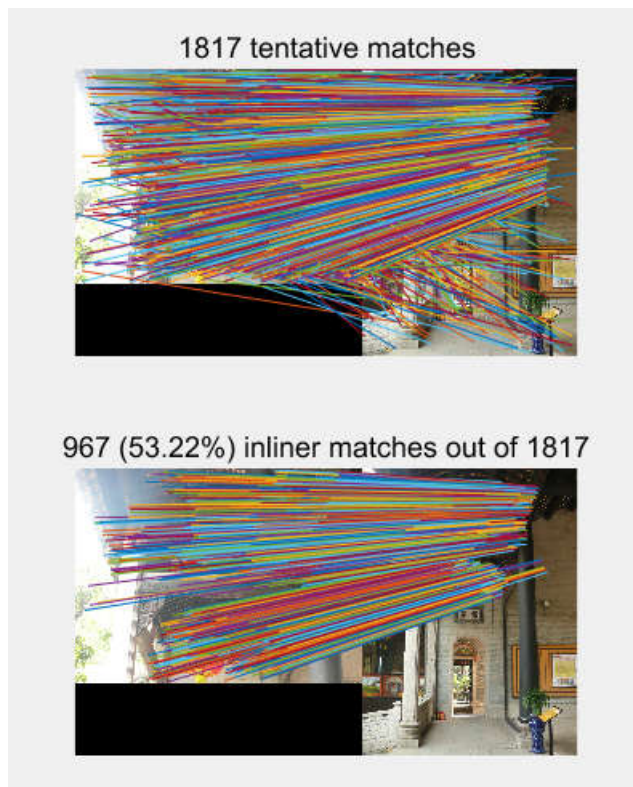
如上图所示，左图为建筑局部图像，右图为建筑整体图像，通过以上方法，左右两幅图像的匹配部分又黄色矩形框标注。观察到匹配区域基本吻合，且主建筑判断正确。

2. 岭南_番禺余荫山房\灯笼

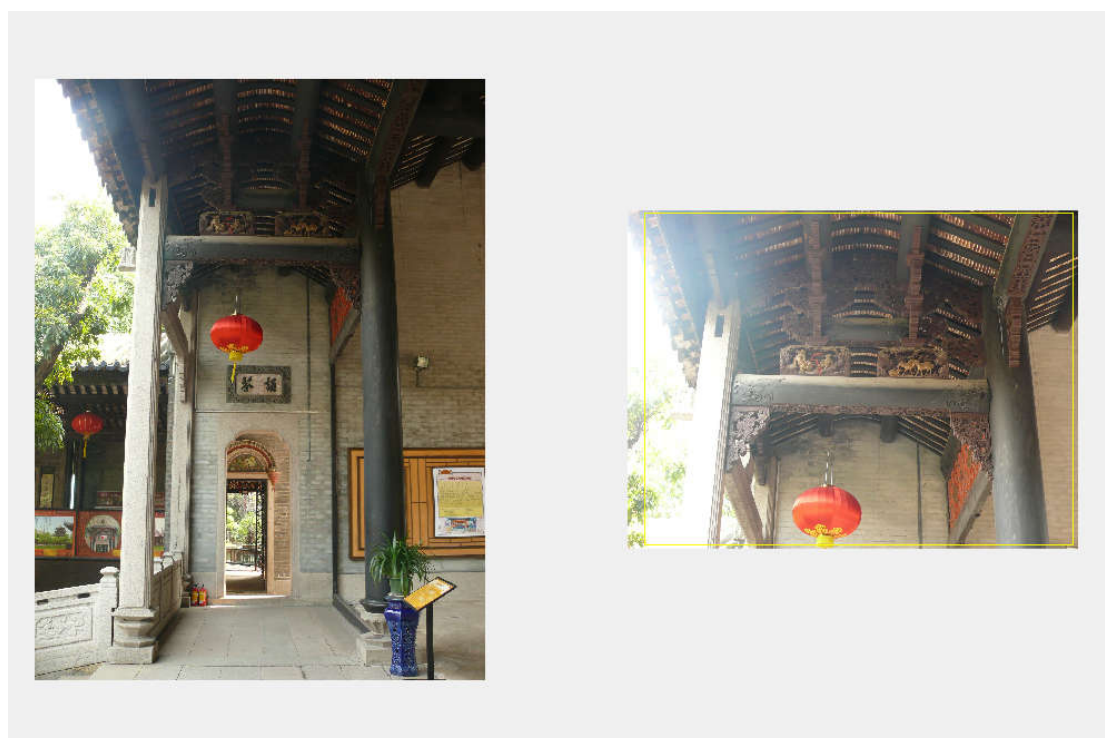
待匹配图像：



筛选匹配点：



匹配结果：



观察如上结果，可知主建筑图像判断不是非常准确，对比两张图像包含建筑信息的程度，本应左边图像被选取为主建筑。分析致使该结果的原因应为图像总数过少，若有足够多的图像作为匹配集，则可得正确结果。但从匹配区域的标注方面来看，采用该方法仍可准确匹配局部信息。

3. 其他例子

岭南_番禺余荫山房\沙湾玉虚宫



三、项目前景

经过此次方法实践，总结出利用 SIFT 匹配图像局部特征并标注的优点与不足之处。

优点：

1. 对于清晰且远近视角差异不大的图像，SIFT 特征提取并匹配的准确率较高，且匹配部分吻合人眼所见的匹配区域。
2. 此方法可用于处理图像间的局部信息匹配和标注，特别针对于某整体图片和取自该图片的局部图片或两相似信息图片的多视角呈现方面。

不足之处：

1. 由于数据集中图像的拍摄情况复杂，加上方法的局限性，对于某些建筑的特征匹配结果不是很理想，尤其是拍摄物的整体与局部细节的视角远近差异很大时。
2. 该方法运行速度较慢，不是非常适合大数据集的处理。