

structure from motion 实验报告

成员

龚逸青：201712809008

方家琦：201712512018

实验环境

- python 3.6
 - opencv-python 3.4.2.16
 - numpy 1.18.1

实现流程

求解 *fundamental Matrix*

- 获取两张图片的 *sift* 特征，获取特征匹配，使用 *ransac* 估计 *fundamental Matrix*。
- 首先随机选出对应点，使用如下方式计算

$$\begin{bmatrix} x'_i & y'_i & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$$

$$x_i x'_i f_{11} + x_i y'_i f_{12} + x_i f_{31} + y_i x'_i f_{12} + y_i y'_i f_{22} + y_i f_{32} + x'_i f_{13} + y'_i f_{23} + f_{33} = 0$$

$$\begin{bmatrix} x_1 x'_1 & x_1 y'_1 & x_1 & y_1 x'_1 & y_1 y'_1 & y_1 & x'_1 & y'_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_m x'_m & x_m y'_m & x_m & y_m x'_m & y_m y'_m & y_m & x'_m & y'_m & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{21} \\ f_{31} \\ f_{12} \\ f_{22} \\ f_{32} \\ f_{13} \\ f_{23} \\ f_{33} \end{bmatrix} = 0$$

通过解上面这个方程得到 *fundamental Matrix*。

- 使用 *ransac* 在有限次的迭代内找到比较有限的基础矩阵。算法内容具体如下：

```

n=0;
for i = 1:M do
    // Choose 8 correspondences,  $\hat{x}_1$  and  $\hat{x}_2$  randomly
    F = EstimateFundmentalMatrix( $\hat{x}_1, \hat{x}_2$ );
    S =  $\emptyset$ ;
    for j = 1:N do
        if  $|x_{2j}^T F x_{1j}| < \epsilon$  then
            S = S  $\cup$  {j}
        end
    end
    if  $n < |S|$  then
        n = |S|;
        Sin = S
    end
end
end

```

算法一 RANSAC求最优基础矩阵

- 上面这个方程可以使用奇异值分解求解最小二乘，假设上述方程为

$$Ax = 0$$

对 A 做奇异值分解， $A = USV^T$ ， V 的最后一列就是我们所求的 *fundamental Matrix*。

上面的方法求出来的矩阵不一定满足秩为2的限制，所以使用 *svd* 分解强制使其秩为2。这样，之后求得的所有极线才能交于同一点。

求解 *essential Matrix*

- 上一步求出 *fundamental Matrix* 后可以通过下面这个公式直接推出：

$$E = K^T F K$$

其中 K 是相机的内参矩阵。

- 由于 *essential Matrix* 的秩也为2，同时还要满足两个特征值的值相等。因此，也使用 *svd* 分解后使其强制满足上述条件。
- 需要注意的是，这里我们求到的 *essential Matrix* 结果不是很优，我个人更推荐使用效果更好的[五点法](#)进行该矩阵的求解。

求解相机参数 R, t

- 对 *essential Matrix* 做奇异值分解：

$$E = USV^T$$

而我們所需要的 t 值为矩阵 U 的最后一列，或是最后一列取负； R 的值为 UWV^T 或者 $UW^T V^T$ 。需要注意的是我们需要保持 $\det(R) = 1$ ，否则，当该值为负时，我们需要对 C 和 R 同时取负。

- 两个变量各有两种取值，共有四种取值。但只有一种取值是正确的，满足实际的投影情况的。我们使用以下方法验证哪一组 R, t 是正确的：

对每一组 R, t 使用三角化求出三维空间坐标，三维空间坐标应满足在两个相机坐标系下 z 坐标的值都不小于0。根据此条件可以对不满足下面公式的异常点（设为 X ）进行计数

$$r3(X - C) > 0 \text{ and } X[3] > 0$$

我们认为异常点最少的一组解往往对应着正确的 R, t 。

三角化

这里稍微提一下，我在实验中首先按照教辅给出的[实验指导](#)进行实现，但是始终没有成功，重建出来的结果往往是一条简单的直线，导致了我在进行 R, t 的筛选时也出了很多错误。可能是在操作中有某些细节没有真正把正负、符号对应关系正确匹配上。在几次尝试未果后，我改用了以下方法：

- 基于上一过程中解得的 R 和 t 以及一组平面对应点 $x = \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix}$ 和 $x' = \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix}$ ，首先对应求得两个相机的pose矩阵 P ：

$$P = KR[I_{3 \times 3} - C]$$

- 再利用最小二乘法求解以下方程：

$$\begin{bmatrix} -P_1[1] + y_1 * P_1[2] \\ P_1[0] - x_1 * P_1[2] \\ -P_2[1] + y_2 * P_2[2] \\ P_2[0] - x_2 * P_2[2] \end{bmatrix} X = 0$$

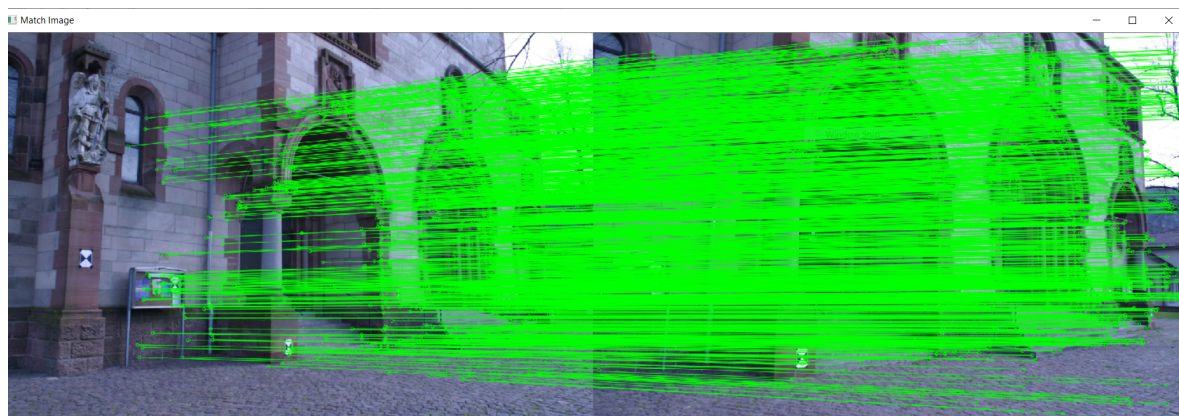
- 得到的结果需要对齐次项进行归一，即每一维除以最后一维的大小，通过这一方法，即可以求得每组correspondences对应的空间坐标，从而通过两张图片恢复出点云。

合并多张图片

实验结果分析

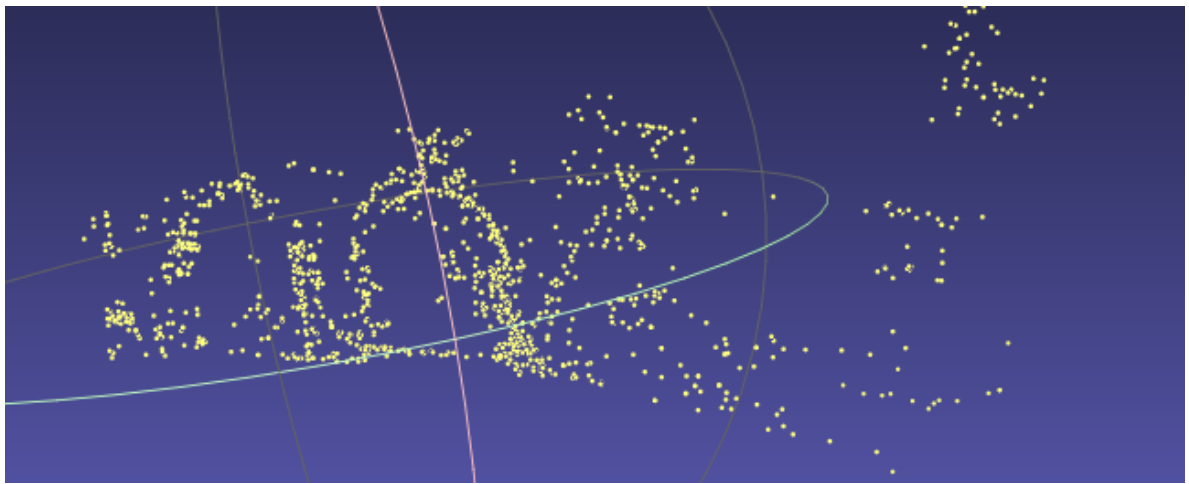
ransac找到的correspondences

如下图一所示，首先给出ransac筛选出的匹配特征点对应关系。就我个人用眼睛的检查而言，效果还是不错的，但在之后的步骤中，这一结果显得并不让人满意。

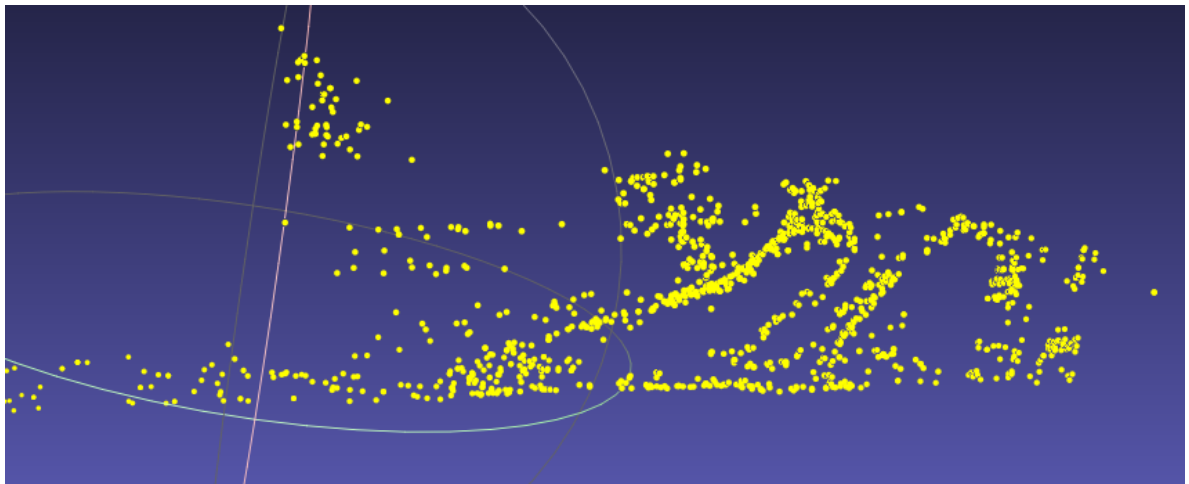


图一 ransac找到的correspondences

第一组图片单独映射的结果



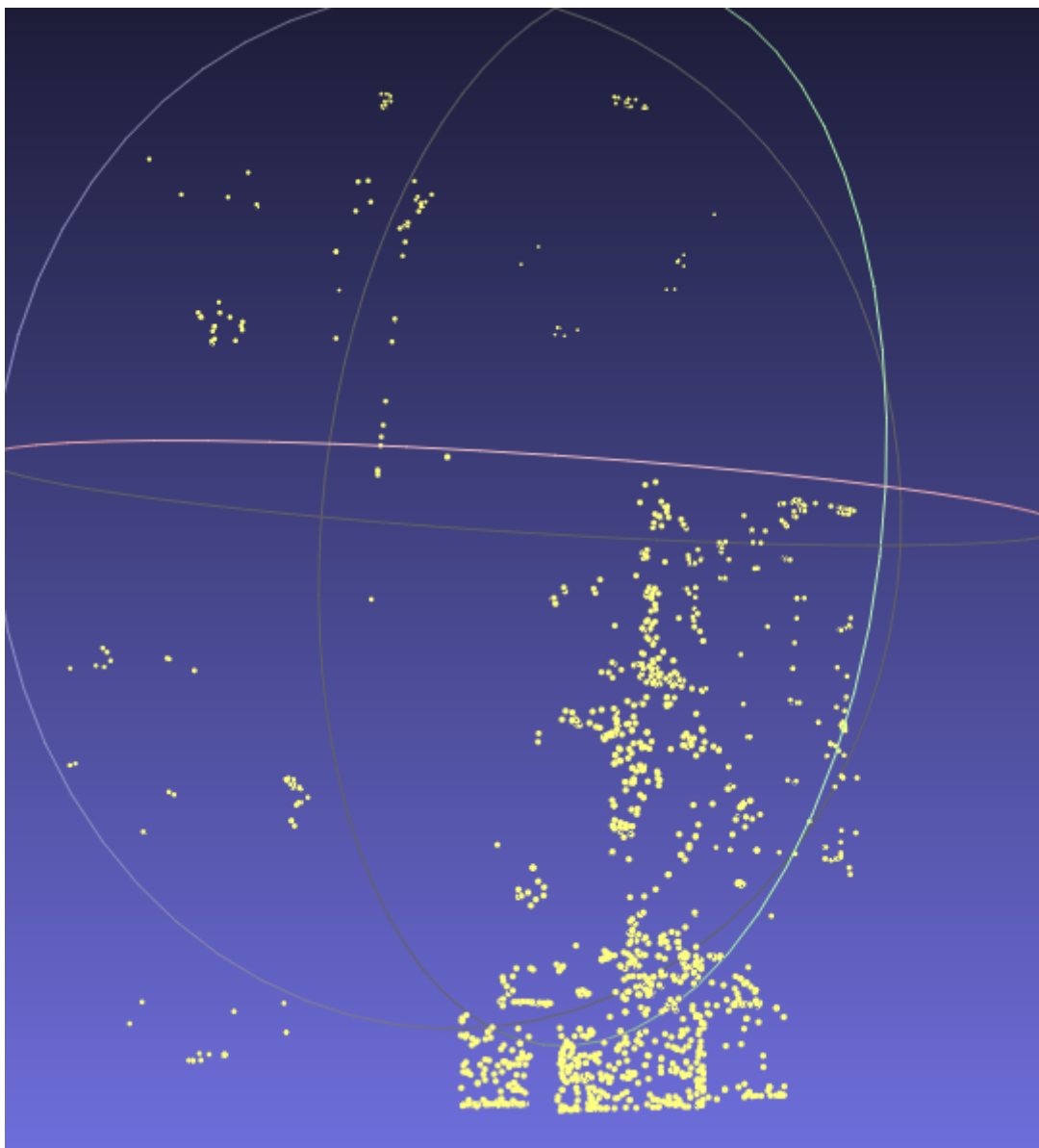
图二 使用cv2库五点法求出的essential Matrix的还原效果



图三 使用我们的方法求出的essential Matrix的还原效果

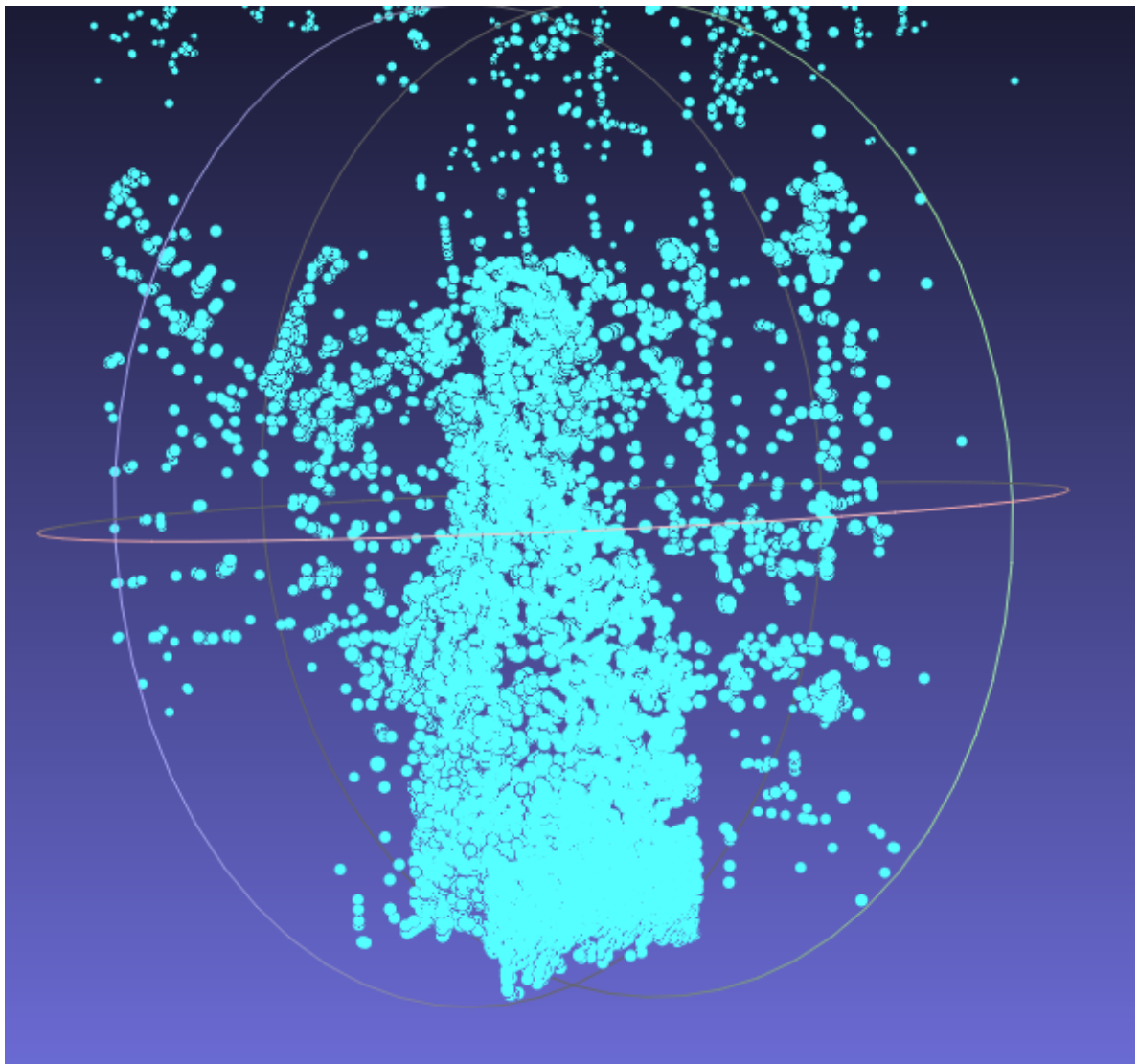
上图二是使用cv2库计算出的 $essential\ Matrix$ 的结果，而图三是使用我们计算出的 $essential\ Matrix$ 得出的结果，大致可以看到轮廓，但是可以看出我们的结果存在一定的扭曲，应该这是由于特征点匹配的效果不好fundamental Matrix的估计存在较大误差所致。

第二组图片单独映射的结果



图四 使用我们的方法求出的essential Matrix在第二组图片上的还原效果

第二组图片映射的全局结果



图五 使用我们的方法求出的essential Matrix在第二组图片上的还原结果

参考文献

[1] [Structure from Motion | CMSC426 Computer Vision](#)

[2] [三角化求深度值（求三位坐标） | michaelhan3](#)