

Task 1 相机模型与对极几何

Task 1-1: 阅读代码文件 `class1_test_math_basic.cc`, 了解矩阵和向量的基本操作与数学运算, 包括矩阵与向量的创建、元素的访问、矩阵向量的乘法、矩阵的奇异值分解等操作。后面的任务中需要用到这些基本的操作。

Task 1-2: 参考 slides 中针孔相机模型部分, 完成 `class1_test_camera.cc` 中相机的 3 个类函数:

```
math::Vec2d projection(math::Vec3d &p3d);  
math::Vec3d pos_in_world();  
math::Vec3d dir_in_world();
```

其中第 1 个类函数是相机的投影过程, 包括世界坐标系到相机坐标系的变换、相机坐标系到归一化像平面的变换、径向畸变、归一化像平面到像平面的变换等流程。具体投影过程为:

世界坐标系到相机坐标系

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \mathbf{R} \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \mathbf{t}$$

相机坐标系到归一化像平面

$$x = \frac{x_c}{z_c}$$

$$y = \frac{y_c}{z_c}$$

径向畸变

$$\tilde{x} = (1 + k_0 r^2 + k_1 r^4) x$$

$$\tilde{y} = (1 + k_0 r^2 + k_1 r^4) y$$

归一化平面到像平面

$$u = f \tilde{x} + u_0$$

$$v = f \tilde{y} + v_0$$

注意 u_0, v_0 为图像中心点坐标, 此处默认为 0 即可。

Task 1-3: 阅读 SIFT 特征匹配代码。

(1) 与特征匹配相关的代码包括:

features/maching.h

features/matching.cc

features/matching_base.h (基类提供接口，有两个子类分别表示两种匹配策略)

features/exhaustive_matching.h (最近邻暴力搜索)

features/exhaustive_matching.cc

features/cascade_hashing(利用哈希表进行加速)

features/cascade_hashing.cc

(2) 填充 matching.h 中函数 oneway_match()函数中关于 lowe-ratio 利用最近邻与次近邻比进行筛选的代码部分。

(3) 完成后测试 examples/task1/test_matching.cc 函数，结果保存在 tmp 文件夹中，比较添加 lowe-ratio 前后的匹配效果。

Task 1-4: 推导并掌握 slides 中直接线性变化法（8 点法），完成 class1_test_fundamental_8_point.cc 中的函数：

```
FundamentalMatrix fundamental_8_point(math::Matrix<double, 3, 8> &points1  
                                     , math::Matrix<double, 3, 8> const & points2)
```

Task 1-5: 掌握 RANSAC 方法的基本原理，掌握 RANSAC 求解基础矩阵的原理和过程，完成 class1_test_fundamental_ransac.cc 中函数

```
int calc_ransac_iterations(double p, int K, double z=0.99);  
std::vector<int> find_inliers(sfm::Correspondences2D2D const & matches  
                             ,FundamentalMatrix const & F, const double & thresh);
```

注意在第二个函数中，判断匹配点是否为内点的标准用的是 Sampson Distance, 定义为：

$$d(x_1, x_2) = \frac{(x_2 F x_1)^2}{(F x_1)_0^2 + (F x_1)_1^2 + (x_2 F)_0^2 + (x_2 F)_1^2}$$

其中 $(F x_1)_0^2$, $(F x_1)_1^2$ 分别表示点的第 0 和 1 维坐标的平方。RANSAC 求解基础矩阵过程中，需要用到 8 点法和最小二乘法估计基础矩阵。最小二乘法估计基础矩阵的方法和 8 点法完全一样，唯一的区别在与 A 矩阵的维度不同。匹配点的信息存储在/examples/task2/correspondences.txt 中，共包含 274 对匹配点。

Task 1-6: 掌握从本质矩阵中求相机参数的方法。

完成 class1_test_pose_from_fundamental.cc 中的相关函数，包括给定基础矩阵，

求解本征矩阵($\mathbf{E} = \mathbf{K}_2^T \mathbf{F} \mathbf{K}_1$), 从本征矩阵中分解得到 4 组旋转矩阵和平移向量, 以及从 4 组 poses 中得到准确的 pose (参见函数 `bool is_correct_pose()`)。注意该函数求得的相机姿态是相机 2 相对于相机 1 的姿态, 而相机 1 的姿态默认为 $\mathbf{R}_1 = \mathbf{I}, \mathbf{t}_1 = \mathbf{0}$, 即第一个相机的相机坐标系与世界坐标系重合。另外, 在选择正确的相机姿态的时候, 用到了三角化(给定两个相机内外参数和一对匹配点, 求解匹配点对应的三维点坐标)的知识, 关于三角化会在下一节中详细介绍。