实验一　机器人抓取对象识别与定位

实验目的：

1、学习基本点云处理方法

2、学习利用模板匹配的方式实现点云工件识别定位

实验设备及软件：

1、相机

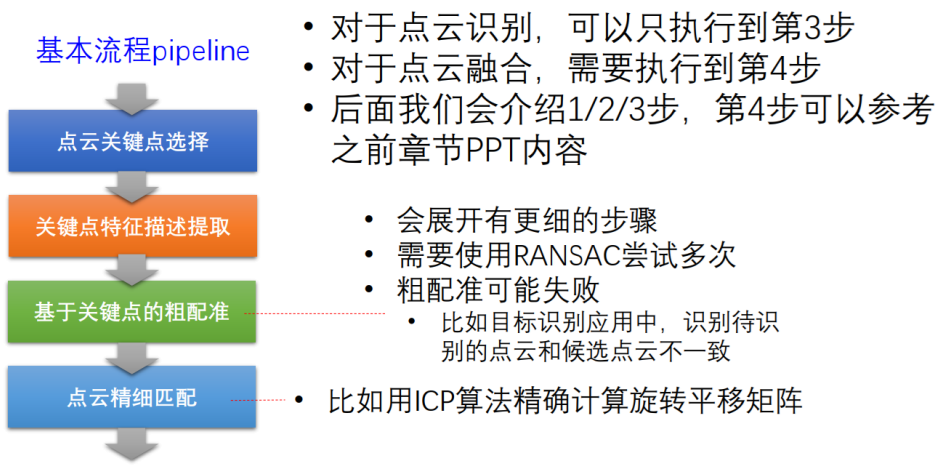
2、Python

实验原理：

基于 3D 点云的模板匹配技术，实现机器人抓取对象识别与定位，从而将目标位姿传输给机器人，提升机器人的智能性。

通过基本点云图像处理方法，实现点云的降采样、平面提取及分割等功能，实现点云预处理。

分割得到的点云与模板进行分别进行关键点与关键点特征描述，实现关键点的粗配准，最后使用 ICP 算法得到精确的旋转平移矩阵。



点云配准(Point Cloud Registration)指的是求得一个变换 ，对于两幅点云 P_s (source) 和 P_t(target) ，使得 $T(P_s)$ 和 P_t 的重合程度尽可能高。对于机器人抓取场景而言，只考虑刚性变换的情况，即变换只包括旋转、平移。

点云配准可以分为粗配准（Coarse Registration）和精配准（Fine Registration）两步。粗配准指的是在两幅点云之间的变换完全未知的情况下进行较为粗糙的配准，目的主要是为精配准提供较好的变换初值（可以用 FPFH+SAC 来做）；精配准则是给定一个初始变换，进一步优化得到更精确的变换（可以用 NDT 和 ICP 来做）。

目前应用最广泛的点云精配准算法是迭代最近点算法（Iterative Closest Point, ICP）及各种变种 ICP 算法。

ICP 算法描述

点云匹配用公式来写就是：

$$R^*, t^* = \underset{R,t}{argmin} \frac{1}{P_s} \sum_{i=1}^{|P_s|} \left|\left|p_t^i - (R \cdot p_s^i + t)\right|\right|^2$$

其中：P_S^i和P_t^i是原始点云与目标点云中对应的点。

ICP 算法的思想如下：

如果我们知道两幅点云上点的对应关系，那么我们可以用 Least Squares 来求解刚性变换 中的 ， 参数；

怎么知道点的对应关系呢？如果我们已经知道了一个大概靠谱的 ，参数，那么我们可以通过贪心的方式找两幅点云上点的对应关系（直接找距离最近的点作为对应点）。

ICP 算法实际上就是交替进行上述两个步骤，迭代进行计算，直到收敛。

ICP 一般算法流程为：

1. 点云预处理

- 滤波（去除一定范围之外的点，去除地面）、清理数据（去除 nan 值）等

2. 匹配

- 应用上一步求解出的变换，找最近点（之前最好先做好粗配准，ICP 对初值的要求比较高）

3. 加权

- 调整一些对应点对的权重

4. 剔除不合理的对应点对
5. 计算 loss
6. 最小化 loss，求解当前最优变换
7. 回到步骤 2. 进行迭代，直到收敛

整体上来看，ICP 把点云配准问题拆分成了两个子问题：

找最近点

找最优变换

实验步骤：

1 设计点云识别定位流程；

2 针对给出的模板点云及目标点云进行点云预处理，并提取关键点及特征描述；

3 匹配得到多个点云工件相对于模板点云的位姿矩阵，并填写下表：

工件序号	位姿矩阵
1	
2	
3	

实验要求：不限方法，不限依赖库，提交实验报告时，请附上 python 程序。