

# 实验七 无人机 SLAM 基础 – 点云信息提取与环境分割

## 1.1 实验目的

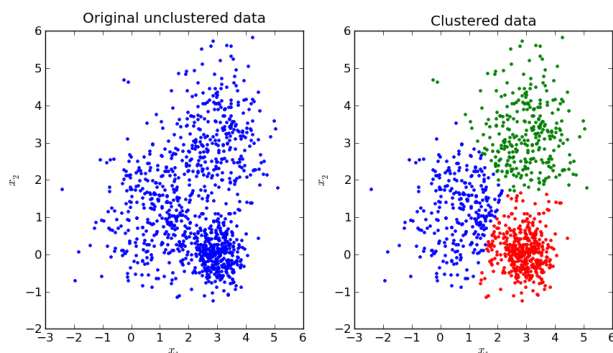
在无人机的各种应用中，测绘作为一种重要功能，已经在近年来得到广大研究者关注。激光雷达作为获取目标散射点三维信息的工具，是无人机在测绘领域应用的重要传感器。

本次实验的目的是在 PCL 库的基础上掌握 C++ 的基本编程，以及基本点云分析、地图构建方法的实现。在前序 PCL 库配置实验的基础上，充分利用 PCL 库以及 Visual Studio 强大的功能对点云进行聚类分析。掌握 K-Means 等方法在点云聚类上的应用，通过点云可视化工具对聚类结果进行分析。在此基础上，掌握算法比较的基本方法。并形成根据实际场景，对算法的优缺点进行辩证分析后选择合适算法的意识。同时，掌握利用 C++ 进行项目开发时必要的编程规范。

## 1.2 实验原理

### 1.2.1 聚类与环境分割

在机器学习算法中，分类和聚类作为有监督和无监督学习的方法，是区分样本的重要手段。聚类是指样本类别信息未知，即不了解一批样本中类别的前提下，对样本进行标记区分的操作。聚类分析方法则是按照某个特定标准，把一个数据集分割成不同的类或簇，使得同一个簇内的数据对象的相似性尽可能大，同时，不在同一个簇内的数据对象的差异性也尽可能大。从而实现“物以类聚”的效果。如下图所示为聚类的实现效果图。



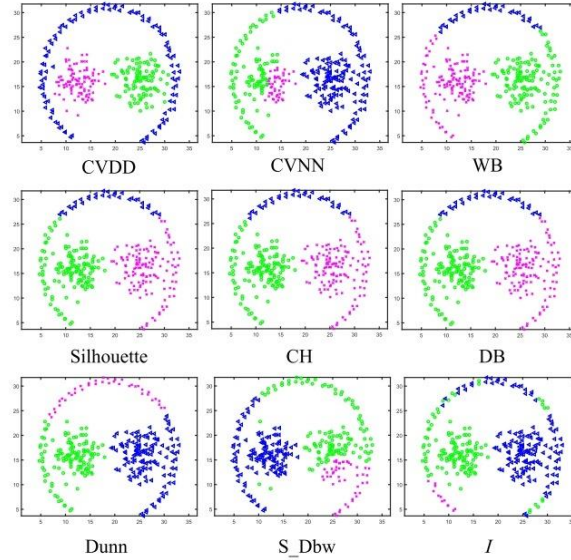
### 1.2.2 算法指标分析与评价方法 – 纯度与混淆矩阵

在聚类算法中，由于样本的标签信息先验未知，完全通过人为计算得出，故衡量指标可以使用误差平方和准则

$$J = \sum_{k=1}^K \sum_{n=1}^{n_k} \|\mathbf{x}_i - \mathbf{x}_k\|_2^2$$

即根据每一轮迭代中，每个簇的所有样本与其聚簇中心的距离和衡量聚类效果。指标分析作为算法之外的计算环节，是衡量算法性能好坏的准则之一。

实际上，由于样本的分布各不相同，如球型分布、带状分布、均匀分布等，如果只采用欧式距离作为聚类的判据，那么对于聚类的效果评价就会受到影响。



欧式距离的评价准则，只适用于样本点没有人假设标签前提下，对样本聚类结果的分析。当样本有人为的类别标记时，纯度作为一个重要的指标，也是衡量聚类结果的重要依据。聚类的纯度用下式表示

$$P = (\Omega, \mathbb{C}) = \frac{1}{N} \sum_k \max_j |\omega_k \cap c_j|$$

其中， $\Omega = \{\omega_1, \omega_2, \dots, \omega_K\}$  为聚类簇样本集合， $\mathbb{C} = \{c_1, \dots, c_J\}$  为类别集合， $|\omega_k \cap c_j|$  表示第  $k$  簇中，类别为  $j$  的样本个数。那么，纯度的计算首先就要先对每个簇中的各类别样本个数进行统计，选取数量最多类样本的数目作为该簇样本对应的主类样本数，将每一簇样本的主类样本数作为正确聚类的情况，并对每一簇进行求和操作，即可达到全局正确聚类的数目。其与总样本数  $N$  的占比即为聚类的纯度。

此外，还有一种衡量聚类效果的方式是二分类混淆矩阵。它也是建立在对于类别信息先验已知的基础上的一种衡量方式。二分类混淆矩阵如下表所示。

判定值\真实值	真(Positive)	假(Negative)
真(True)	TP	TN
假(False)	FP	FN

可以看出，其中 TP 和 FN 的占比越大，积累效果越好。其中，兰德系数(Rand Index - RI)是用于衡量聚类结果的一个重要指标，其计算如下

$$RI = \frac{TP + TN}{TP + FP + FN + TN}$$

RI 的取值范围为[0,1]，值越大说明聚类效果越好。

实际上，对于多分类的聚类方法，还可以用二分类混淆矩阵的物理意义去计算其分类效果。TP 用于表示两个样本点在同一个簇中的情况；TP+FP 用于表示同一个簇中，任意取两个样本点的所有情况；TP+FN 用于表示任意两个同类样本点分布在同一个簇和非同一个簇的所有情况总和，TP+TN+FP+FN 表示所有样本中选取两个样本点的情况。若根据上述定义，则需要计算相应的组合数，以表示兰德系数，统计聚类结果

### 1.2.3 K-Means 聚类分析方法

作为一种经典的聚类分析方法，K-Means 方法的基本思想是先计算各个簇的质心信息，通过比较样本空间中每个样本点与质心的距离，选取最近的质心作为该样本的簇。通过迭代更新实现聚类。

假设  $M$  个  $N$  维空间中样本点  $(\mathbf{x}_i, i=1,2,\dots,M)$  共分为  $K$  簇  $(k=1,2,3,\dots,K)$ ，记作  $\mathbf{x}_i = (x_1, x_2, \dots, x_N | k_i)^T$ 。其中，每一簇的质心记为  $\mathbf{x}_k (k=1,2,3,\dots,K)$ ，上标<sup>(0)</sup>表示当前迭代的次数，则  $K$  均值聚类算法的实现过程大致如下：

#### ① 数据预处理

该部分主要针对实际的数据分布情况，对数据进行初步处理，从而方便聚类的进行。常见的数据预处理方法主要包括噪声去除、离群点去除等。此外，还需要根据实际数据的分布情况，对样本空间进行合理划分，从而合理选择每个簇的初始质心。

#### ② 初始化

$K$  均值聚类算法是一种无监督学习方法，样本的标签信息未知，因此需要事先对样本的标签进行假设。首先，需要随机寻找样本空间中的  $K$  个点作为各个簇的质心，记为  $\mathbf{x}_k^{(0)} (k=1,2,3,\dots,K)$

#### ③ 距离计算与比较

在随机选好  $K$  个点作为每个簇的质心后，需要对所有样本点进行标记，即质心需要“吸取”周围的点，将其归为自己所在的类。其中，一个重要的归类依据就是各个样本点到所有质心点的欧氏距离。欧氏距离的定义如下

$$d_{ik} = \|\mathbf{x}_i - \mathbf{x}_k\|_2$$

其中， $\|\cdot\|_2$  表示向量的二范数，即其与原点的距离。

$$\|\mathbf{x}\|_2 = (\mathbf{x}^T \cdot \mathbf{x})^{\frac{1}{2}}$$

对于  $M$  个样本中的每一个点  $\mathbf{x}_i$ ，选取其距离最近的质心作为标签

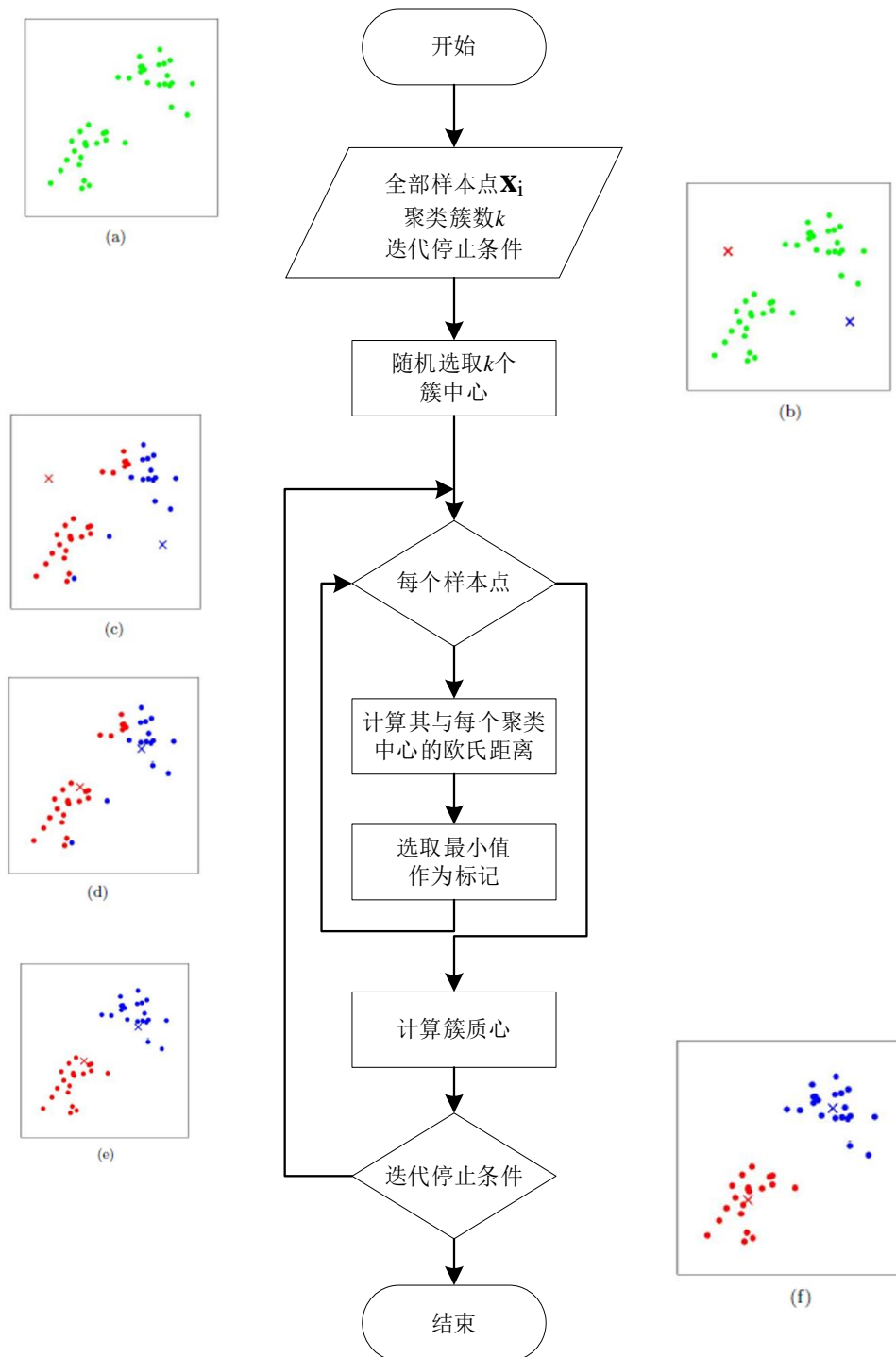
$$k_i = \arg \min_k d_{ik}$$

④ 计算各簇的质心

经过一轮“吸收”之后，各簇的聚类中心需要重新计算。根据每一类点在各个维度的均值，即可得到各个簇下一次迭代的质心。

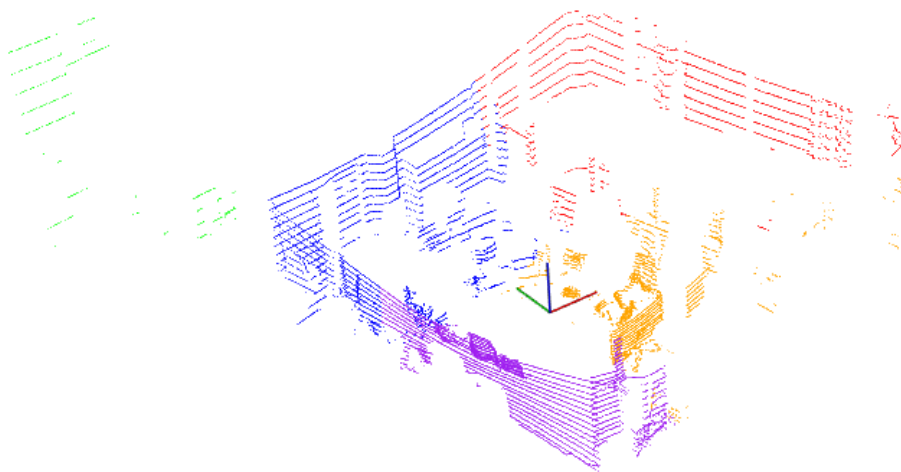
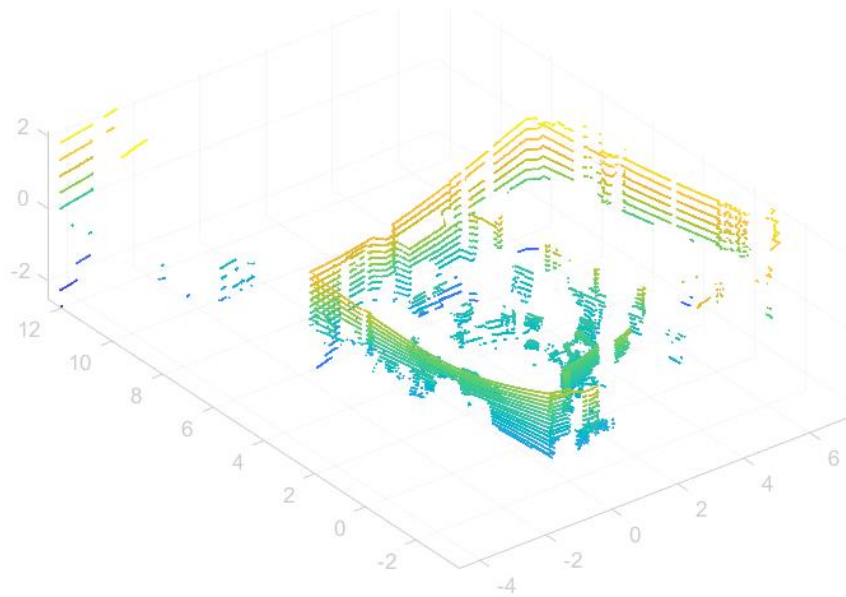
⑤ 重复步骤③，④，直到达到停止条件。

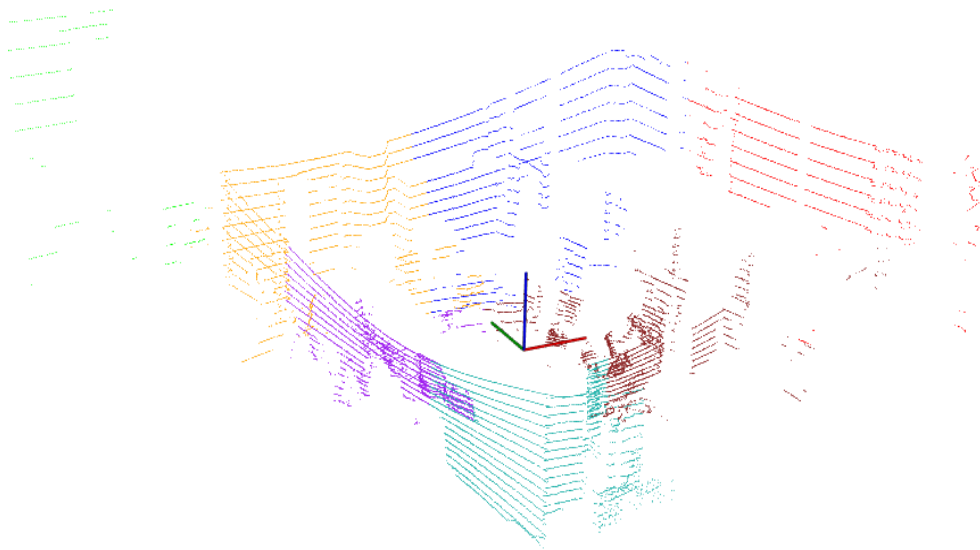
如下图所示为 *K-Means* 算法的流程图以及原理图



### 1.3 实验要求

根据某室内无人机实录点云数据，使用 PCL 库对点云进行初始化。通过对点云对象的点进行分析，建立数据样本。在读取样本的基础上，编程进行 K-Means 聚类，其中，K-Means 需要封装成方法。调试并运行程序，分析运行结果及错误原因。实现对点云的 K-Means 聚类，将点云显示在图窗中并统计聚类准确度指标。





## 2 实验过程

1. 利用 PCL 库建立视窗显示点云信息
2. 结合示例程序，在 C++ 工程中建立头文件，头文件名称为 KMeans\_学号  
后三位.h，同时创建相应的 cpp 格式同名源文件，将 K-Means 封装成类
3. 根据原理 1.2.1 部分的流程图，对流程图进行细化。设计可以用于算法实  
现的子函数，并将其声明在写在头文件中，函数体写在源文件中。
4. 根据声明的子函数，结合流程图和 KMeans 聚类原理补充函数
5. 输入点云数据，调试并运行程序。
6. 根据调试出现的错误，分析错误原因。
7. 若调试成功，编写程序显示聚类后的点云，并统计聚类准确度。

## 3 附录 – 见附件参考程序

