

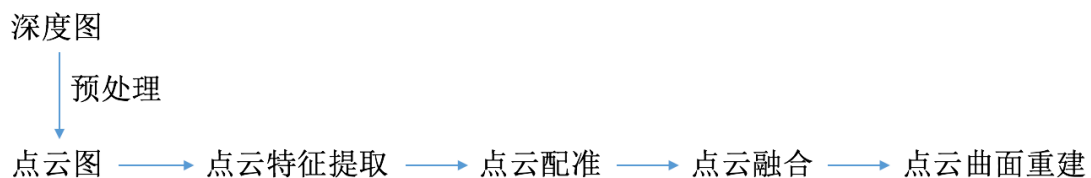
基于深度图的三维重建

一、评价指标

方法的复杂程度、实时重建完成度、重建的精细程度以及适用场景。其中重建的精细程度没有具体的评价指标，目前主要是通过人眼来观察，判断真实物体和重建模型之间的相似性。近年来有一些论文中总结了一些评价方法，如重建误差法、积分误差法、三维基尼系数法等，这些评价方法并不能准确反映三维重建的好坏，需要结合一下会有效果，目前只有对比各种方法的论文，没有发现整合好的方法，这一方面我还会继续寻找。

二、技术路线

下图的技术路线是我根据PCL官网介绍的路线整理出来的。目前只整理到通过深度图进行重建的部分。



1.图像获取：直接使用Kinect进行获取。mac使用<https://github.com/OpenKinect/libfreenect>提供的库来连接kinect。

2.深度图像的预处理：

(1) 分割：由于不同距离的物体在深度图中的灰度值不同，所以直接采用阈值分割即可。阈值分割的代码十分容易。

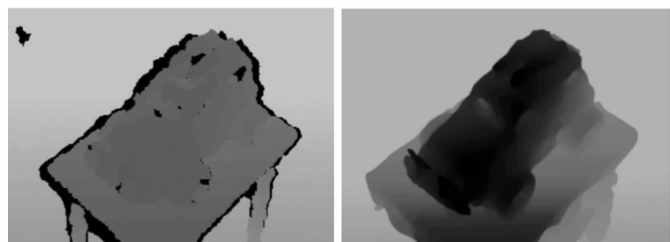
(2) 滤波：可以使用PCL自带的函数进行简单的滤波。迭代联合多边滤波比自带的函数效果好很多。方法来自Lai P L, Tian D, Lopez P. Depth map processing with iterative joint

multilateral filtering[C]// Picture

Coding Symposium. IEEE,

2010:9-12.。右图为例，示例来自于郭连朋, 陈向宁, 徐万朋,等. 基于

kinect传感器的物体三维重建[J]. 四川兵工学报, 2014(11):119-123.。



(a) 原始深度图像

(b) 预处理后的深度图像

3.点云的匹配、融合、重建

(1) 深度图变点云图：OpenNi自带的函数ConvertProjectiveToRealWorld() 和 ConvertRealWorld ToProjective() 直接可以进行简单的转换。

(2) 点云特征提取、配准、融合：

代码方面：PCL官网<http://www.pointclouds.org/documentation/tutorials/#range-images>上以及我在图书馆下载的《点云库PCL学习教程》光盘资料中有介绍和详细代码，目前我还在归类。

算法方面：研究的比较多的是配准方面的论文，配准过程中最经典的算法是ICP，现在已经有很多基于ICP的优化算法。俞老师让我在配准过程中可以借鉴SLAM中的有关技术。

(3) 点云曲面重建：

计算机视觉里用到的比较多的是隐式曲面重建方法，郑老师推荐了一篇Michael Kazhdan的《Poisson Surface Reconstruction》就是用的隐函数的方法，在Michael Kazhdan的主页<http://www.cs.jhu.edu/~misha/Code/>中介绍了很多重建的其他方法。

三、benchmark（目前我想先完成简单的重建，从重建的过程中发现有没有可以改进的地方）

滤波方面：Lai P L, Tian D, Lopez P. Depth map processing with iterative joint multilateral filtering[C]// Picture Coding Symposium. IEEE, 2010:9-12.介绍了一种处理深度图像的迭代联合多边滤波方法。

点云配准方面：俞老师推荐的SLAM技术中的配准过程，SLAM有关技术的教程我目前找到《SLAM for Dummies》这一本，这本书的末尾有一些伪代码，可以根据它的代码思路进行不一样语言的移植，或者是用来理解其中的一些过程。

点云的曲面重建方面：郑老师推荐了一篇文章《Poisson Surface Reconstruction》是关于泊松表面重建，就是运用了隐函数的方法。计算机视觉里用到的比较多的就是这种隐式曲面重建法。

四、研究方案

1.获取深度图像和对应彩色图像

我们使用Kinect进行图像的获取。Kinect可以快速进行物体的扫描，获得深度图像和对应的彩色图像。由于它价格低、体积小、扫描速度快，目前已经被广泛应用于软件开发和小场景的三维重建。

Kinect的使用方法（基于macos）：由于mac并不能直接运行微软官方提供的Kinect CDK，所以我们只能尝试使用非官方破解的软件在mac上进行安装。Hector Martin发起的库可以用来运行kinect，下载地址：<https://github.com/OpenKinect/libfreenect>，安装方法以及需要的软件都在网站中有详细的介绍。

2.使用获取的深度图像进行三维建模

2.1 图像预处理

由于设备本身的限制，采集到的图像不可避免的出现噪声和一些不利于处理的缺陷，我们需要对采集到的图像进行滤波、增强等预处理。为了更好地进行匹配，我们还需要对图像进行分割操作。

2.2 基于深度图像的匹配、融合、重建

由于图像只包含二维信息，我们想要完成三维重建必须需要三维信息，这就需要点云图的支持。我们主要使用PCL（Point Cloud Library，点云库）来进行操作，这也是使用人数最多的，公认最完备、最优秀的点云处理编程库。PCL包括很多先进的算法和典型的数据结构，如滤波、分割、配准、识别、追踪、可视化、模型拟合、表面重建等诸多功能，能够在各种操作系统和大部分嵌入式系统上运行。在算法方面，PCL是一套包括数据滤波、点云配准、表面生成、图像分割和定位搜索等一系列处理点云数据的算法。PCL的使用要依赖于很多包和软件。下载地址为：https://www.pointclouds.org/documentation/tutorials/compiling_pcl_macosx.php。下载方法都已在网站给出详细的操作流程。PCL十分强大，它几乎包含了所有可以在三维重建过程中用到的基本算法，而且全部都有开源代码供测试。开源代码库可以在<http://www.pointclouds.org/documentation/tutorials/#range-images>上找到。

想要深入研究匹配过程中的算法，我们首先可以对匹配过程的算法进行研究：粗匹配和精细匹配。我们在得到点云图以后，需要对点云图进行粗匹配，粗匹配是在完全不知道对应关系的情况下，寻找两图对应特征点大致区域的过程。由于点云处理数据众多，这样做可以大大减少后续匹配中花费的时间。精准匹配在完成粗匹配之后进行，精准匹配是为了找到精准的对应特征点。精准

匹配现在主要使用迭代就近点法ICP (Iterative Closest Point) 或者它的改进算法。ICP算法是由Besl和McKay在1992年发表的文章《Method for registration of 3-D shapes》中提出的，这种算法不仅仅考虑了点集与点集之间的配准，还有点集到模型、模型到模型的配准等，现在还有很多对于这种算法的创新方法。另外，在网站<https://www.cnblogs.com/yhlx125/p/4955337.html>上有对算法介绍的各种论文，可以加深对算法的理解。点云匹配解决的多张照片之间的特征点匹配问题，有了点云匹配的结果我们就可以进行点云融合。点云融合将多幅图像的点云数据进行融合从而获得完整的点云模型。根据完整的点云融合模型就可以进行曲面重建。在曲面重建中的算法也值得研究，目前计算机视觉采用的比较多的曲面重建方法是隐式曲面重建方法，这里面可以有很多的算法进行改进和学习，比如RBF、MPU等。