**点云处理相关技术调研2021初步**

**0调研记录（持续更新中）**

By kyanko：

2021年9月13日，对点云数据独有的一些特性进行归纳总结；

2021年9月24日，增加对FPFH的调研，学习了利用AxMath进行公式编辑，相关文件后续上传；

2021年12月1日，增加对Harris3D算法调研，并进行相关公式编辑；

2022年3月7日，增加ISS3D算法调研，并进行相关公式编辑；

2022年3月9日，对ISS3D公式进行符号修正；

2022年4月13日，专业方向变动，调研暂停，欢迎后续补充。

By Haraton：

2022年4月17日，增加点云去噪相关算法调研整理；

2022年4月22日，修改部分公式并重绘

**1引言**

3D点云与2D图像不同，它包含了更加丰富的坐标信息，也相对于2D图像隐藏了更多空间结构信息与更丰富的语义信息，处理点云数据并获得其隐含的各种信息便显得尤为重要，在计算机技术高速发展的现代，随着计算手段的增加和计算效率的大幅提高，处理海量点云数据也变的更加现实，各种点云特征提取算法与点云关键点检测算法也得到了高速发展。本文档对其中的部分经典技术进行了调研与汇总，便于各位同学交流学习。

本文档目前阶段主要分为三个部分，第一部分主要针对点云几何特征提取算法进行相关调研，第二部分针对点云关键点检测算法进行调研，第三部分对点云去噪相关算法进行调研。

**2点云几何特征提取**

点云中包含着丰富的几何信息，几何特征提取的目标就是要识别并检测点云中局部邻域中的几何结构，通过将邻域中的几何结构信息编码为表示为向量的特征空间中的点，借助式2.3所描述的几何特征相似假设，进而可以通过分析特征空间中的近邻点来获取对应关系的集合。几何特征提取是点对配准中最为关键也是最为复杂的内容，因为特征的好坏直接关系到特征匹配的结果，进而影响刚性配准的结果。一个好的几何特征必须对离群值、噪声、异常点和点云密度的变化鲁棒，并且需要具备较高的特征辨识度。在点云上提取几何特征的方法可以区分为深度学习方法和传统人工编码的方法，由于深度学习的飞速发展，基于深度学习的几何特征提取算法已经大幅优于传统的人工编码的方法。本小节重点介绍基于深度学习的几何特征提取算法。

点云是一种描述空间点的位置、颜色等信息的数据结构，相比于图像这种规则的二维数据结构，点云具有无序性、稀疏性、信息不完整性和无组织性等特点。为了能够在点云上提取高辨识度的几何特征，上述点云的四个特点是主要要解决的问题。在本小节中，首先详细介绍点云的四个特性，然后再对具体的解决方案进行介绍。

1.无序性

与二维的图像中的像素矩阵不同，点云是一组没有特定顺序和排列的点的集合。具体来说，点云通常的表示形式是一个N×3的矩阵，N表示点云中点的个数，矩阵的每一行是点的三维坐标。无序性的具体表现在于，交换矩阵任意两行，此点云是不变的。由于使用不同传感器或从不同角度对同一物体进行扫描得到的点云其点与点之间的顺序是变化的，并且一个包含*n*个点的点云模型，可能有*n*!个不同顺序的排列。因此无序的点云对深度学习的影响较大，可能导致网络过拟合，影响网络训练的效果。现阶段应对点云无效性的方法主要是通过数据增强的方法去训练网络，即在网络训练的每一个epoch中对点云数据进行随机地重排列。通过这种方式可以有效防止因为点云的无序性导致的网络过拟合的问题。

2.稀疏性

由于传感器的限制，三维扫描设备获取的点云通常只位于物体的表面，并且物体的形状对获取的点云也有影响。例如，激光雷达在近处扫描得到的点云较为密集，而在远处的点云较为稀疏。对于RGBD相机Kinect，场景中包含一些黑色的物体可能会导致扫描的点云中包含一些空洞。通常来说，点云越密集，那么对于物体表面的描述更为精准。点云越稀疏，效果相反，这会显著影响几何特征的辨识度，并且由于两者在局部区域中包含的点数不同，这可能导致求取的几何特征显著不同。对于点云的稀疏性，通常可以使用体素化降采样来解决，但是虽然体素化降采样可以调整密度，但是也会降低点云的分辨力，造成几何特征的辨识力下降的问题。

3.信息不完整性

点云是一组三维空间点坐标构成的离散点集。由于本质上是对三维世界中物体几何形状进行低分辨率重采样，因此点云数据提供的几何信息是不完整的；另外，点云数据采集时由于遮挡等原因，无法获取目标物体完整的三维描述。点云的信息不完整性对几何特征提取提出了更高的要求。

4.无组织性

点云中包含丰富的几何拓扑信息，但是点云是无组织结构的数据。类似二维图像，可以通过像素矩阵的形式对逐点进行组织，点云也可以表示为一个N×3的矩阵，但是这样的组织方式过于简单，不利于网络学习点云的几何拓扑结构，因此需要一些更利于表现点云几何信息的数据结构来组织点云。体素是一种常用的组织结构，体素是一个单位的立方体包围盒，通过体素化可以将点云分别用小的立方体包围起来，得到多个立方体。通过体素化，可以方便地使用深度神经网络模型进行特征提取。但是这种方法由于内存限制,只能使用比较小分辨率的体素网格，并且体素包围盒的边缘部分的点云拓扑结构被破坏，造成信息的丢失。

上述点云数据的特性导致了在处理点云数据时不能像传统二维图像上应用卷据操作一样。为了能够挖掘点云中丰富的几何拓扑信息，提取更为丰富的几何特征，近几年研究人员进行了深入的研究和探索，本文将现有方法大致分为了两种：基于局部块的方法、基于点云的方法。基于局部块的方法指的是通过体素、包围球、最近邻等方法将空间中距离较近的点组织到一起得局部块的方法。基于点云的方法是直接将点云作为输入通过网络的自学习来捕获点云中的局部几何结构的方法。下面对这三类方法中的一些经典的方法进行介绍。

**2.1 快速点直方图**

FPFH是一种几何特征描述符，考虑局部范围内所有点之间的位置影响和法线关系，描述局部范围内数据的几何特征，具有位置信息不变性的特点。FPFH是点特征直方图（Point Feature Histograms，PFH）的扩展，主要在计算效率方面进行了较大改进，但是依然保留了PFH大部分的特性，计算复杂度为O(nk)。

快速点特征直方图的计算过程如下：首先确定中心点P与邻域点之间的法线偏差。法线偏差可以用法线之间的角度值进行表示，该步骤的结果为简化的点特征直方图（Simplified Point Feature Histogram，SPFH）。然后，根据相同的方法，查找中心点所有邻域点的邻域范围，计算每个邻域点邻域范围内的SPFH，则中心点P的FPFH的计算方式：

 (1)

其中，*ωk*表示权值，可以利用中心点距离邻域点的空间距离来近似表示权值。如图9所示，表示中心点**P**在计算FPFH过程中的影响范围。虚线内表示中心点**P**的邻域范围，其邻域点集为Pn = {**Pn1**, **Pn2**, **Pn3**, **Pn4**, **Pn5**}，根据**P**与邻域点的法线偏差，计算得到点**P**的SPFH(**P**)。然后，分别确定邻域集合Pn内所有点的邻域范围，即图中五个彩色区域对应点集Pn内五个点的邻域，计算点**P**的邻域点**Pnk**的SPFH(**Pnk**)，最后利用公式求得FPFH(**P**)。

图示

描述已自动生成

图2-1 中心点P计算FPFH的影响范围

**2.2 曲率特征**

超体素数据的边界精确性会直接影响最后分割结果的边界精确度。为了确保过分 割得到的超体素数据具有精确的边界信息，需要在过分割过程中考虑三维体素数据的 边缘信息。

图片包含 游戏机, 照片, 桌子, 房间

描述已自动生成

图2-2 点云数据的曲率表示

三维体素数据的边缘信息往往对应着三维物体中曲面弯曲程度较大的区域，可以 用曲率来衡量表面的弯曲程度。因此，本文提出的过分割算法在特征计算及距离度量的过程中，通过利用曲率来表示三维体素数据的边界属性，确保过分割得到的超体素 可以依附于物体的边缘信息，具有精确的分割边界。

**3点云关键点检测**

关键点检测是点云配准技术中的一个关键环节。关键点不仅包含了模型的重 要信息，而且有助于减少点云计算的数据量，可以节约计算时间和存储空间。

早期大部分三维关键点检测算法都是直接从二维拓展而来，如SIFT（Scale Invariant Feature Transform）、Harris等算法。近年来，关键点检测算法得到快速 发展，2009年Yu Zhong提出了ISS（Intrinsic Shape Signatures，固有形状特征）关 键点检测方法，该方法稳定、可重复、信息丰富且具有识别力，能够确保高精度 的3D形状匹配和识别。2010年Radu B S等人提出了NARF（Normal Aligned Radial Feature）关键点检测和特征描述方法，该方法明确使用对象边界信息，并试图在表面稳定但附近有显著变化的区域提取特征。2015年Filipe Silvio等人受到 灵长类视觉系统的行为和神经架构的启发，提出了一种基于自上而下的3D显著性图提取的关键点技术。2018年AC Bernal等人采用稀疏关键点检测技术进行目标检测，从方向表面计算出关键点，而方向表面是用对应于均匀表面的法向量建立的。2019年Muhammad Zafar Iqbal等人提出了一种新的三维点云关键点检测器， 该算法利用点云数据的3D结构和RGB信息，通过计算点云结构信息的协方差矩 阵的特征值来生成候选关键点，然后，通过基于高斯算子的有效自适应差分，从RGB信息中估计显著点，最后将得到的两组显著点进行融合，以提高3D关键点 检测器的可重复性。2020年Lu Fan等人提出了一种基于随机样本的关键点检测 器和描述符网络（RSKDD-Net，Random Sample-based Keypoint Detector and Descriptor Network）算法。该算法的关键思想是使用随机抽样来有效地选择候选 点，并使用基于学习的方法联合生成关键点和描述符。2021年Shi Ruoxi等人提 出了一种无监督对齐的关键点检测器——骨架合并，利用骨架来重建物体，该算法基于自动编码器架构，编码器提出关键点并预测关键点之间边缘的激活强度，解码器对骨架执行均匀采样，并将其细化为具有逐点偏移的小点云，然后应用激活强度合并子集。

目前存在的关键点提取算法有很多，各种算法有不同的优缺点，根据各算法 的计算原理，其在模型上对关键点检测时无论是数量还是旋转平移不变等特性均 有不同。因此，研究关键点检测算法针对模型特征的适应性，对于点云配准是很 有实用性的。

**3.1 Harris3D**

Harris2D算法是由Chris Harris等人提出的一种组合的角点和边缘检测器（Harris corner detector），该检测器使用图像上的一个局部窗口在图像上进行任意方向上的移动，比较移动前后窗口中图像灰度的平均变化程度。若任意方向移动，灰度的平均值都有较大强度变化，则认为该窗口内存在角点。一般只需考虑三种情况：

（1）窗口在平坦区域移动（如：灰度近似常数），引起较小变化；（2）如果窗口跨越边缘，沿边缘移动会引起较小变化，垂直边缘方向的移动会引起较大变化；

（3）如果窗口内存在角点或孤立点，则沿各个方向移动，平均灰度均会引起剧烈变化。

若用I表示图像灰度，E表示由移动（m，n）引起的变化，则用数学公式表达上述变化为：

 (2)

其中，w为所选窗口，（x，y）是窗口中像素坐标。根据泰勒公式一阶展开：

 (3)

代入得到：



 (4)

将中间的矩阵用M代替，得到：

 (5)

其中，*M*为2×2矩阵，*Ix*，*Iy*是在*x*和*y*方向的导数。如果将*M*的特征值记为*α*和*β*，*α*和*β*将与局部自相关函数的主曲率成正比，并形成对*M*的旋转不变描述。则之前三种情况可更改为：

（1）特征值*α*和*β*都很小，窗口在平坦区域移动。

（2）特征值*α*和*β*，一个特征值较大，另一个特征值相对较小（理想情况下等于0），此时窗口中存在边缘；

（3）特征值*α*和*β*都很大，此时窗口中含有角点。

用特征值分类角点或边缘区域很方便，但用其来测量角点和边缘质量或响应并不是很方便。因此，Chris Harris等人提出了一个角点响应函数*R*，其计算公式如公式所示：

 (6)

其中，*Det(M)*=*αβ，Trace(M)=α+β*，*k*是选取的控制参数。在角点区域*R*为正值，在边缘区域*R*为负值，在平坦区域*R*很小。

将该方法扩展到3D领域，方向强度变化通过计算中心点的法线与其相邻点之间的两个垂直方向的一次微分来替代。

如图3所示，Harris3D以当前所取的点v为中心，在周边设置一个环形范围，采到该范围内的点集（如：从绿色点至黄色点处），并基于这些点的分布拟合得到一个二维曲面。然后在这个曲面上的两个相互垂直方向上(设置为*x*和*y*方向)进行一次微分的梯度计算，分别计为*fx*，*fy*。

图表, 雷达图

描述已自动生成

图3-1 Harris3D计算面

由微分构建*M*矩阵：

 (7)

其中，

 (8)

 (9)

 (10)

这里的*σ*表示高斯函数的方差范围。在得到*M*矩阵之后，即可以根据公式计算角点响应函数值。

**3.2 ISS3D**

ISS（Intrinsic shape signatures，固有形状特征）关键点提取方法是由Yu Zhong在2009年时提出的，它是一种表示3D模型几何形状的方法。ISS由一个固有的参考框架和一个对3D形状特征进行编码的高度区分的特征向量组成，该参考框架能够实现视图不变特征提取和快速姿态配准。

对于三维点云中的任意一点*P，*在其支撑半径*r*内，计算该点的协方差矩阵，然后通过矩阵的特征值分解求取特征值*λ*1，*λ*2，*λ*3，且满足*λ*1>*λ*2>*λ*3。

 (11)

其中，*Pk*为点*P*在半径*r*内的所有邻域点，由于3D点采样时并非是均匀采样，为了让稀疏采样区域的点比密集采样区域的点贡献更多，采用权*wk*来补偿。接着，设置阈值参数*γ*12和*γ*23，值一般要小于1。在修剪阶段，留下两个连续特征值之比低于阈值的点作为候选关键点，如式所示：

 (12)

该限制旨在满足可重复性，并避免在沿主方向具有相似分布的点处检测关键点。在NMS非极大值抑制阶段，显著性由最小特征值的大小确定，以便仅包括沿着每个主方向都变化很大的点。在检测步骤之后，如果某个点在给定邻域中具有最大的显著性值，则该点将被视为关键点。

**4点云去噪算法**

激光点云的噪声分为离散噪声点和混合噪声点两类。离散噪声在空间上远离目标点，混合噪声在空间上和目标点混合在一起。对于离散噪声点，主要是利用噪声点在空间位置上和目标点存在着明显的差异进行噪声的去除，已经形成的方法有：基于统计的去噪算法，半径滤波去噪算法，基于聚类的去噪算法。

对于基于统计滤波的去噪方法，Kalogerakis等人提出了一种利用点云表面法向和曲率的统计估计来去除点云噪声的方法。刘静静提出了一种基于统计特征改进的离群点去噪算法。首先根据统计分析的方法计算出每个点的最优k邻域，并且首先去除稀疏的离散点。然后根据最优邻域半径去除其他的离散噪声点。

为了比较统计滤波和半径滤波两种方法的去噪性能，鲁冬冬等人将统计滤波算法和半径滤波算法进行对比，表明在点云中离散点远离主体点云，且位置分散时，半径滤波能够更加快速的去除点云中的噪声点，但是对于和目标主体混合在一起的噪声点难以有效的去除。对于混合噪声点，主要是利用曲面平滑的方式进行去除，已经形成的方法有：基于主成分分析算法去噪，基于偏微分方程的曲面逼近算法和双边滤波算法。Mattei提出了一种基于运动稳健主成分分析的点云数据去噪算法，能够对不具有明显特征的点云中包含的与目标混合在一起的混合点噪声进行滤除。但是这种方法对于具有高噪声水平和减税特征的点云会增加算法运行的时间，并且由于算法估计的法线是通过主成分分析获得的，它们的方向不一致，这需要在后面的处理步骤中重新定位输出法线，造成了算法实现的困难。Lozes等人提出了一种基于偏微分方程对点云去噪滤波的方法，这种方法对于具有尖锐特征的点云不能够很好的保持点云的特征。Fleishman等将应用在图像处理的双边滤波方法引入到三维点云网格平滑去噪中,提出各向异性保特征去噪算法。这种算法在尖锐的目标边缘容易导致经过若干次迭代得到的平面会去掉尖锐部分，不能够较好的保留边缘。

为了将离散噪声点和混合噪声点同时去除，Wang等人提出了一种有有效的去噪算法，主要分为两步：离散噪声点去除和噪声平滑。他们首先设计了一种基于连通性的算法去消除稀疏的离散点，接着使用了一种基于聚类的方法消除聚集的离散点，最后提出了一种基于稳健部分排序的噪声点法线估计算法去平滑噪声。这种方法取得了不错的去噪效果。但是，由于点云在边缘的部分比较稀疏，因此可能有一部分的真实点被检测成异常值，造成了错误去除。另外，对于室外包含复杂结构的场景，例如树木，使用这种方法不能很好的保持特征。现有的点云去噪算法主要利用了激光点云的空间信息，在空间上将噪声点和目标点进行区分。但是现有的点云去噪算法存在局限性：一方面，在目标的边缘和尖锐部分，真实点和噪声点在空间上的差异不明显，仅仅利用空间信息难以有效的去除噪声点；另一方面，由于离散噪声点和混合噪声点具有不同的空间分布特点，现有的单色的去噪算法无法将离散噪声点和混合噪声点的去除方法统一，造成这种方法的步骤繁琐。

**4.1 现有去噪方法**

(1)基于统计的去噪方法。基于统计的去噪方法是利用统计知识对点云的某些特征进行统计分析，从而计算出每个点是噪声点或者是非噪声点的概率大小，从而决定是否对该点进行去除或者保留。Digne提出了一种根据点云局部位置的相似性来对噪声进行去除的算法。Jenke基于贝叶斯统计的方法，将采样点以及先验知识建模为概率分布，通过求出最大的概率值进行去噪。这种算法能够比较好的保持点云细节特征。但是这种方法对于尖锐目标边缘部分的噪声难以有效去除。

(2)半径滤波。半径滤波是根据真实点在空间分布上相对集中，离散噪声点在空间分布上相对离散的原理提出的。如图所示4.1所示，半径滤波首先设置一个搜索半径，然后以某一点为中心，根据给定半径画一个圆并统计落在该圆内点的数量。最后根据圆内点的数量判断某一点是否为噪声点，从而决定保留还是剔除。

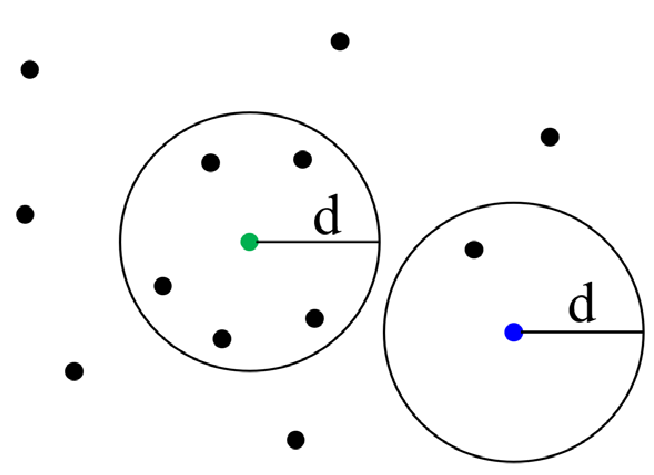


图4-1 半径滤波示意图

(3)基于聚类的去噪方法。基于聚类的去噪算法主要是根据点云的空间分布情况来进行聚类去噪的，比较典型的有基于密度空间聚类算法及其改进算法、模糊C均值聚类算法、K means聚类等。基于密度的聚类算法最典型的算法就是DBSCAN算法。如图2-7所示，这种算法首先以每个数据点为圆心，以eps为半径画一个圆，数据点的数量就是该点的密度值；接着取一个密度阈值Min Pts，根据每个数据点的密度值和密度阈值Min Pts的大小关系将数据点分为高密度点和低密度点。如果有一个高密度点在另一个高密度点的圈内，就把这两个点连接起来；如果有低密度点也在高密度点的圈内，把这个低密度点和高密度点连接起来。这样所有连接到一起的点就形成了一个簇，而不在任何高密度点的内的低密度点就是噪声点。

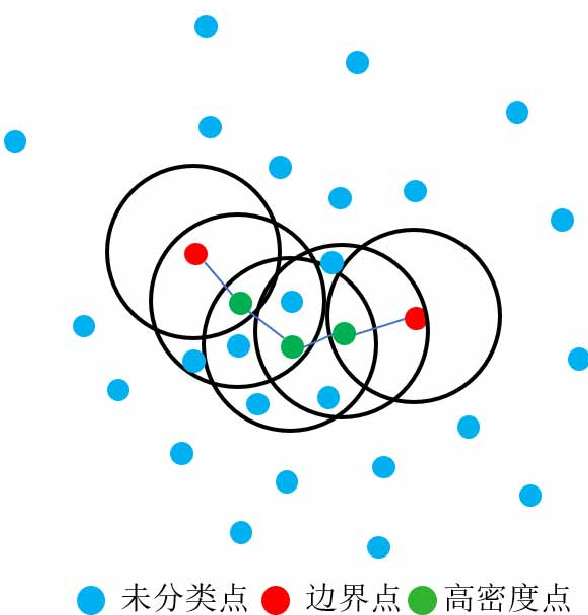


图4-2 DBSCAN聚类示意图

DBSCAN聚类算法对于点云数据中孤立噪声点的去除效果较好，但是对于不均匀数据，例如目标的边缘部分，这种去噪算法的效果不理想。