

# 平面参数空间的实时三维点云配准方法<sup>①</sup>

王力宇<sup>②</sup> 曹其新<sup>③</sup> 王雯珊

(上海交通大学机器人研究所机械系统与振动国家重点实验室 上海 200240)

**摘 要** 针对具有多平面结构的室内环境的三维定位和环境建模,提出了一种在平面参数空间进行配准的实时三维点云配准方法。该方法首先使用一种改进的三维霍夫变换算法快速提取点云中的平面特征,然后使用迭代算法在平面的参数空间中寻找最近平面,最后使用这些平面的对应关系来估计两帧点云之间的位姿变换关系。在一个实验室场景中进行的对比实验表明,该算法能够达到与传统迭代最近点(ICP)算法相似的精度,而且速度大大提升。在仅使用普通笔记本 CPU 的情况下即可实现实时的点云拼接。

**关键词** 点云配准,平面提取,霍夫变换,三维地图重建,迭代最近点(ICP)

## 0 引 言

激光点云地图在机器人定位及导航、环境探索、灾难救援等方面具有广泛的应用。目前 2D 激光点云地图的创建问题已经较为成熟,已广泛应用于自动导引车(AGV),扫地机器人等移动平台。然而利用 3D 传感器的地图创建问题仍然是研究热点。传统的点云配准算法主要有迭代最近点(iterative closest points,ICP)算法<sup>[1]</sup>和正态分布变换(normal distribution transform,NDT)算法<sup>[2]</sup>两种。

ICP 算法的核心思想是通过寻找两帧点云之间欧氏距离最近的点,根据这些点对求解两帧点云的位姿关系,然后利用该位姿关系对点云进行位姿变换,通过不断迭代上述过程求解最终的位姿变换<sup>[1]</sup>。而 NDT 算法的核心思想是将目标点云栅格化,然后根据每个栅格中点云的空间分布,使用三维正态分布函数来拟合点云。然后将待配准点云的坐标值经过平移  $t$  和旋转  $R$  后代入到该正态分布函数中,使用迭代优化函数求解  $R$  和  $t$ ,以使该正态分布函数的取值最大<sup>[2]</sup>。然而 3D ICP 和 NDT 算法都需

要对大量数据点进行迭代计算,耗时较长。因此通常都需要使用 GPU 加速才能实现实时的 3D 点云配准<sup>[3,4]</sup>。使用 GPU 的算法对电脑性能要求很高,应用场景受限。

另一类基于 3D 点云的实时地图拼接与创建需要借助 RGBD 传感器,使用较成熟的 RGB 图像的特征点来进行匹配,然后根据 RGB 图像与深度图像的对应关系求解位姿变换<sup>[5]</sup>,但是现有的 RGBD 传感器测量范围大都十分有限(Microsoft Kinect 一代约 4m<sup>[6]</sup>,Microsoft Kinect 二代约 4.5m<sup>[7]</sup>,ASUS Xtion Pro 约 3.5m<sup>[8]</sup>,Intel RealSense 长距离版约 4m<sup>[9]</sup>),因此这些方法都无法用于比较空旷的环境,这也限制了这些算法的应用范围。

室内环境通常具有较多的平面结构,由于平面特征的稳定性,很多研究者尝试使用平面特征来加快速点云配准的速度和增强配准的稳定性。Weingarten 等人使用扩展卡尔曼滤波算法对提取出的平面进行配准<sup>[10]</sup>,但是他们的算法需要提供卡尔曼滤波的预测值。这可以融合惯性测量单元(IMU)及里程计的值得到,也可以预先对点云进行 ICP 配准,使用该配准结果作为预测值,但是前者需要机器人配备

① 国家自然科学基金(61273331)资助项目。

② 男,1991 年生,硕士生;研究方向:机器人定位和导航,机器人运动控制;E-mail: wangliyu@sjtu.edu.cn

③ 通讯作者,E-mail: qxcao@sjtu.edu.cn

(收稿日期:2016-11-22)

昂贵的 IMU 传感器,而后者则加大了运算量。Pathak 等人首先定义了两个点云平面匹配的四个条件,然后使用这些条件对两帧点云中的所有平面一一检测,找出匹配的平面。最后根据检测到的匹配平面,计算出两幅点云的平移和旋转关系<sup>[11]</sup>,他们的方法能够在有较多干扰的环境下保持较高的配准精度,但是使用上述枚举方法寻找匹配平面较为耗时,无法达到实时配准的要求。Taguchi 等人对检测到的平面的点进行重采样,并结合点云中的其他稳定的特征点,对这些点进行配准,由于舍掉了点云中的那些不稳定的点,相比于体素滤波(voxel grid filter)<sup>[12]</sup>等随机重采样方法,他们的方法可以使用更低的重采样率保持相似的精度<sup>[13]</sup>。肖博文等人则根据提取出的点云平面的多边形外形进行配准,实现了对于岩体点云的较为精确的配准<sup>[14]</sup>,但他们对于相似外形的多边形的查找也是通过枚举的方法实现,对于配准速度的提升也比较有限。

本文在利用霍夫变换提取平面的基础上,使用迭代算法对这些平面的参数空间进行配准,寻找点云的变换关系。由于使用了迭代算法取代枚举方法进行平面配准,因此该算法能够实现快速的平面配准,能够使用普通 CPU 实现接近实时的 3D 点云配准算法。

## 1 基于霍夫变换的平面特征提取

在点云中提取平面特征主要有三种方法:基于随机采样一致性(random sample consensus, RANSAC)的方法<sup>[15]</sup>,基于霍夫变换(Hough transform)的方法<sup>[16]</sup>和区域增长法<sup>[17]</sup>。本文使用文献[18]中提出的基于改进霍夫变换的点云平面提取方法,该方法使用八叉树(Octree)和主成分分析(principal components analysis, PCA)来加快霍夫变换,能够实现实时的点云平面提取。

### 1.1 平面的参数空间表示

霍夫变换使用公式

$$\hat{n}^T p - \rho = 0 \quad (1)$$

中的 Hessian 参数方程将平面  $P: Ax + By + Cz + D = 0$  转换到  $(\theta, \phi, \rho)$  的参数空间。式中  $p =$

$[x \ y \ z]^T$  为平面上的点坐标,  $\rho \in [0, \infty)$  为平面与原点的距离。 $\hat{n}$  为平面的单位法向量,其展开式如下:

$$\hat{n} = \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = \begin{bmatrix} \cos(\theta) \sin(\phi) \\ \sin(\theta) \sin(\phi) \\ \cos(\theta) \end{bmatrix} \quad (2)$$

其中,  $\theta \in [0, 2\pi)$  为该法向量与  $x$  轴的夹角,  $\phi \in [0, \pi]$  为该法向量与  $z$  轴的夹角。

通过在参数空间投票,可以求出参数空间中的极值点,即为点云中的平面。

### 1.2 改进的霍夫投票算法

传统的三维点云霍夫变换需要对每个点进行投票,由于三维点云数据量比较大,因此投票和极值求解的过程需要消耗较长的时间,对于一帧含有约  $10^4$  个点的点云,进行一次霍夫投票大概需要数十秒的时间。为了加快投票速度,我们使用了一种改进的霍夫投票算法,使得霍夫投票的时间缩短到了 0.1s 左右,能够满足实时性的要求。

该方法首先对点云进行八叉树分割,然后对于分割后的每个八叉树节点进行主成分分析(PCA)<sup>[19]</sup>,找出该点云的两个单位主方向向量  $\hat{d}_1$ 、 $\hat{d}_2$  以及与其正交的单位方向向量  $\hat{d}_3 = \hat{d}_1 \times \hat{d}_2$  以及这些方向向量对应的特征值  $\lambda_1$ 、 $\lambda_2$  和  $\lambda_3$ 。

主方向向量的特征值  $\lambda_1$ 、 $\lambda_2$  与其正交方向向量的特征值  $\lambda_3$  的比值  $\lambda_3/\lambda_1$  和  $\lambda_3/\lambda_2$  可以用来表示该八叉树节点的平面度。如图 1 所示,当点云分布接近球形分布时,  $\lambda_3/\lambda_1 \approx 1$ ,  $\lambda_3/\lambda_2 \approx 1$ ; 当点云分布接近平面时,  $\lambda_3/\lambda_1 \leq 1$ ,  $\lambda_3/\lambda_2 \leq 1$ ; 当点云分布接近直线(或柱面)时,  $\lambda_3/\lambda_1 \leq 1$ ,  $\lambda_3/\lambda_2 \approx 1$ 。因此,我们可以通过选择阈值  $\delta_1$  和  $\delta_2$ , 当  $\lambda_3/\lambda_1 < \delta_1$  且  $\lambda_3/\lambda_2 < \delta_2$  时,我们认为该八叉树节点为平面。经过实验测试,当  $\delta_1 = 0.04$ ,  $\delta_2 = 0.15$  时,平面检测效果较好。

使用以上条件来判断各八叉树节点的平面度,如果该节点的特征值比值满足阈值,则认为找到了一个小平面,停止分割,并将方向向量  $\hat{d}_3$  视为该小平面的法向量。否则继续分割,直到特征值的比值满足阈值或者八叉树节点的剩余点云数量  $n$  小于阈值  $n_p$ 。该阈值的选取与输入点云的密度有关,本文

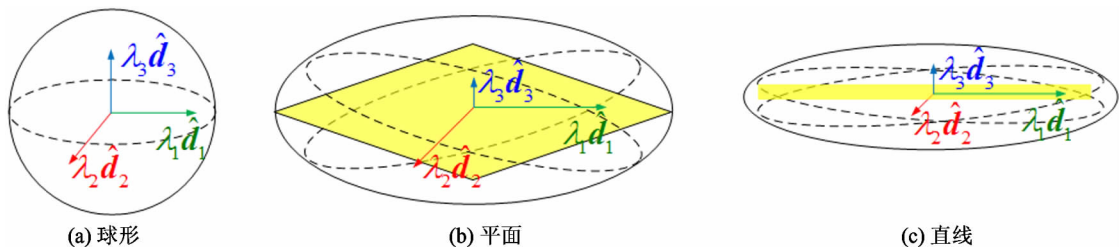


图1 对应不同形状特征的方向向量及其特征值

将该阈值取为 20。对一幅点云的八叉树分割结果如图 2 所示。通过八叉树分割,本研究将原本包含数万个点的点云变成了只包含数百个八叉树节点的集合,极大地减少了后续霍夫投票和极大值求解过程的运算量。

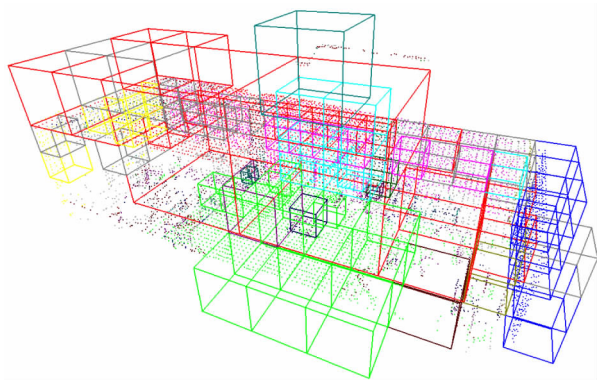


图2 点云的八叉树分割结果

然后使用基于高斯核的霍夫变换<sup>[16]</sup>求得每个八叉树节点的投票值,每个节点的权重使用该八叉树节点的体积  $V_i$  和其中包含的点数  $n_i$  计算,即对于每个八叉树节点,其权重为

$$w_i = \omega_v \frac{V_i}{V_c} + \omega_n \frac{n_i}{n_c} \quad (3)$$

其中:

$V_c$  表示点云的外接立方体的总体积;

$n_c$  表示点云中的总点数;

$\omega_v$  为体积比的权重,它表示如果该八叉树节点的体积越大,则其投票时权重越高;

$\omega_n$  为点数比的权重,它表示如果该八叉树节点包含的点数越多,则其投票时权重越高。

实验发现,适当提高  $V_i/V_c$  的权重更有助于获得稳定的平面提取效果,在本文中,设置  $\omega_v = 0.75, \omega_n = 0.25$ 。

最终的投票结果反映在以  $(\theta, \phi, \rho)$  为球坐标的霍夫球上。霍夫投票的结果如图 3 所示,在该图中,本研究使用色温来表示投票值的大小,暖色温表示投票值较大,冷色温表示投票值较小。

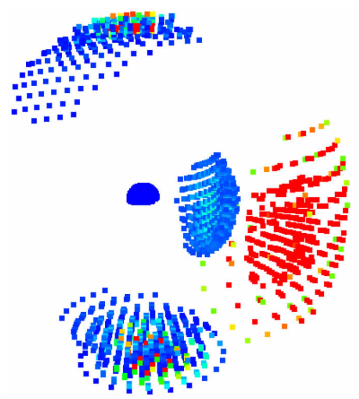


图3 霍夫投票的结果

最后对投票值大于一定阈值的点求极大值,寻找点云中的平面。本文选择的阈值为所有投票值的中位数。极大值选取方法与常规的 3D 霍夫投票的极大值选取方法一致<sup>[20]</sup>,即若一个点的投票值大于与其相邻的 26 个点的投票,则认为该点为极值点,该点的参数代表了一个平面。在一副点云中提取的平面如图 4 所示,其中不同的颜色代表不同的平面。

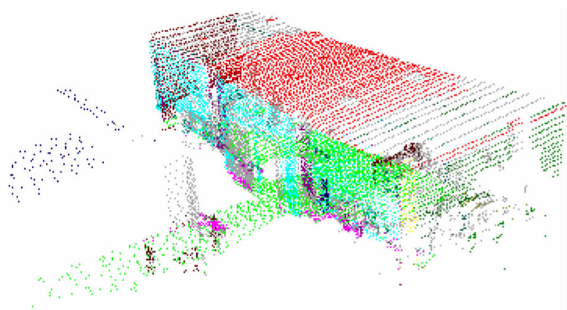


图4 在一副点云中找到的平面

## 2 参数空间中的平面配准

如图 5 所示,三维空间中的一个点  $P$  可以限制一个物体的三个方向上的平移自由度,三维空间中三个不共面的点可以完全确定一个物体的位姿。类似地,三维空间中的一个面  $A$  可以限制一个物体的一个平移自由度和两个旋转自由度,三维空间中 3 个不平行的平面可以完全确定一个物体的位姿。

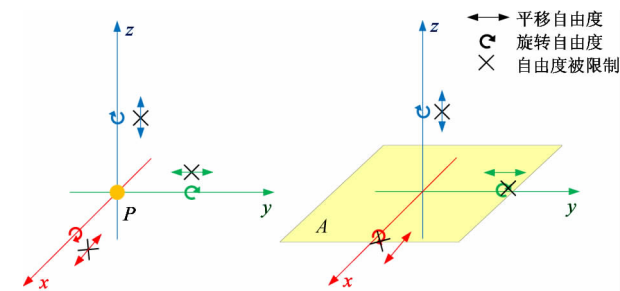


图 5 三维空间中点和平面限制的自由度

通过霍夫变换,本文可以获取到点云中的平面参数。因此,本研究可以在参数空间中直接对这些平面进行配准,利用平面的配准结果来估计两帧点云的位姿关系。

### 2.1 平面的平移和旋转

本文使用 Hessian 参数方程来表示平面方程,使用旋转  $R$  和平移  $t$  将平面  $P$  转换为平面  $P'$ ,即平面  $P$  上的任意点  $p$  通过旋转  $R$  和平移  $t$  转换为平面  $P'$  上的点  $p'$ :

$$p' = Rp + t \quad (4)$$

此时的平面方程变为

$$\hat{n}'^T p' - \rho' = 0 \quad (5)$$

将式(4)带入式(5),并与式(1)比较系数,同时考虑到  $R^T R = I$ ,可得,变换之后的平面  $P'$  的法向量  $\hat{n}'$  和原点到平面距离  $\rho'$  和原平面  $P$  的法向量  $\hat{n}$  和原点到平面距离  $\rho$  之间的变换关系为

$$\begin{aligned} \hat{n}' &= R\hat{n} \\ \rho' &= \hat{n}'^T t + \rho \end{aligned} \quad (6)$$

### 2.2 参数空间中的对应平面查找

要想进行平面配准,首先需要找到两个点云中的对应平面。

本文将平面的法向量  $\hat{n}$  与原点到平面的距离  $\rho$  相乘,得到点  $v = \rho\hat{n}$ 。该点的几何意义是原点到该平面的垂线与该平面的交点。根据式(6),当通过旋转  $R$  和平移  $t$  把平面  $P$  变换到平面  $P'$ ,此时原点到该平面的法线与该平面的交点变为

$$v' = ((R\hat{n})^T t + \rho) R\hat{n} \quad (7)$$

点  $v'$  与点  $v$  的欧氏距离为

$$\begin{aligned} d &= |v' - v| = |(R - I)v + R\hat{n}(R\hat{n})^T t| \\ &\leq |(R - I)v| + |t| \end{aligned} \quad (8)$$

因此,当平移和旋转比较小时,可以认为在参数空间中平面  $P$  和  $P'$  对应的点即为使  $v$  和  $v'$  欧氏距离最短的点,且该距离小于一定的阈值  $\sigma$ 。因此对于点云  $C$  中的任意平面  $P_i$ ,其在点云  $C'$  中的平面集合  $\{P'_1, P'_2, \dots, P'_m\}$  中的对应平面为

$$\begin{aligned} \{P'_{ik} \mid d_{v'_{ik}-v_i} = \min(|v'_j - v_i|) \text{ 且 } d_{v'_{ik}-v_i} < \sigma\} \\ v'_j \in \{v'_1, v'_2, \dots, v'_m\} \end{aligned} \quad (9)$$

阈值  $\sigma$  的选择与 ICP 算法中内点 (Inliers) 选择的阈值<sup>[1]</sup>一致,它与两帧点云之间机器人的可能的位移和旋转量有关。例如若机器人在采集两帧点云时的最大位移量不超过 0.5m,最大旋转角度不超过  $10^\circ$ ,离机器人最远的平面不超过 4m,则

$$\sigma = 0.5 + 4 \times \frac{10}{180}\pi \approx 1.2\text{m} \quad (10)$$

### 2.3 参数空间中的平面配准

平面配准的目的是要计算旋转  $R$  和平移  $t$ ,使得对于点云  $C$  中的平面集合  $\{P_1, P_2, \dots, P_n\}$  和点云中  $C'$  的对应平面  $\{P'_{1k}, P'_{2k}, \dots, P'_{kk}\}$ ,都满足

$$\begin{aligned} \hat{n}'_{ik} &= R\hat{n}_i \\ \rho'_{ik} &= \hat{n}'_{ik}^T t + \rho_i \end{aligned} \quad (11)$$

为了获得两帧点云之间最终的位姿变换  $R$  和  $t$ ,本文使用迭代的方法来求解。

首先在当前的位姿下,寻找两帧点云在参数空间中的对应点  $v_i$  和  $v'_{ik}$ ,并根据其对应关系求解单次迭代结果下的旋转  $\bar{R}$  和平移  $\bar{t}$ 。

根据式(11),对于每组对应的  $v_i$  和  $v'_{ik}$ ,可以得到

$$[\hat{n}'_{1k} \ \hat{n}'_{2k} \ \dots \ \hat{n}'_{kk}] = \bar{R}[\hat{n}_1 \ \hat{n}_2 \ \dots \ \hat{n}_k] \quad (12)$$

因此,当两个参数空间中的对应平面  $k \geq 3$  时,可以求得  $\bar{R}$  的最小二乘解。由于旋转矩阵  $\bar{R}$  是正交矩

阵,因此可以使用奇异值分解来求解  $\bar{\mathbf{R}}^{[21]}$ 。

求出  $\bar{\mathbf{R}}$  后,由式(11)可得其平移  $\bar{\mathbf{t}}$  满足

$$\begin{bmatrix} (\mathbf{R}\hat{\mathbf{n}}_1)^T \\ (\mathbf{R}\hat{\mathbf{n}}_2)^T \\ \dots \\ (\mathbf{R}\hat{\mathbf{n}}_k)^T \end{bmatrix} \bar{\mathbf{t}} = \begin{bmatrix} \rho'_{1k} - \rho_1 \\ \rho'_{2k} - \rho_2 \\ \dots \\ \rho'_{kk} - \rho_k \end{bmatrix} \quad (13)$$

这是一个普通的三元超定方程组,因此可以使用 Moore - Penrose 广义逆来求解平移  $\bar{\mathbf{t}}$  的最小二乘解。

然后使用  $\bar{\mathbf{R}}$  和  $\bar{\mathbf{t}}$  对点云  $C'$  中的平面集合进行位姿变换,对于变换后的平面集合,再次在参数空间中寻找其最近点,直到找到的最近点之间的旋转矩阵  $\bar{\mathbf{R}}$  接近  $\mathbf{I}$ ,平移向量  $\bar{\mathbf{t}}$  的二范数接近 0。

在参数空间中进行平面配准的总体流程如下:

输入: 目标点云  $C$  中的平面参数集合

$$V = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\}$$

待配准点云  $C'$  中的平面参数集合

$$V' = \{\mathbf{v}'_1, \mathbf{v}'_2, \dots, \mathbf{v}'_m\}$$

平面参数空间的最大距离阈值  $\sigma$

迭代终止阈值  $\varepsilon$

输出: 点云  $C$  到  $C'$  的旋转变化  $\mathbf{R}$  和平移变化  $\mathbf{t}$

初始化:  $\mathbf{R} \leftarrow \mathbf{I}; \mathbf{t} \leftarrow \mathbf{0}$

对集合  $V$  建立  $k-d$  树

过程: 重复

根据式(6)使用  $\mathbf{R}$  和  $\mathbf{t}$  将  $V'$  中的所有点  $\mathbf{v}'_i$  变换为  $\mathbf{v}''_i$ , 组成集合  $V''$

对任意  $V''$  中的点  $\mathbf{v}''_i$

在  $k-d$  树中查找最近点  $\mathbf{v}_{ik}$

如果  $d_{\mathbf{v}_{ik}-\mathbf{v}''_i} < \sigma$

找到一个对应点组  $\langle \mathbf{v}''_i, \mathbf{v}_{ik} \rangle$

对所有对应点组  $\langle \mathbf{v}''_i, \mathbf{v}_{ik} \rangle, i = 1, 2, \dots, k$

根据式(12)求解  $\bar{\mathbf{R}}$

根据式(13)求解  $\bar{\mathbf{t}}$

$$\mathbf{R} \leftarrow \mathbf{R}\bar{\mathbf{R}}$$

$$\mathbf{t} \leftarrow \mathbf{t} + \bar{\mathbf{t}}$$

直到  $\|\mathbf{R} - \mathbf{I}\|_2 < \varepsilon$  且  $\|\mathbf{t}\| < \varepsilon$

### 3 实验

针对本文提出的算法,本研究在实验室搭建的 3D 环境建模平台上进行了测试,如图 6 所示。它包含了一台 SICK LMS511 单线激光传感器,一个用于旋转 SICK 单线激光传感器获取三维点云的激光转台以及一台 AGV。

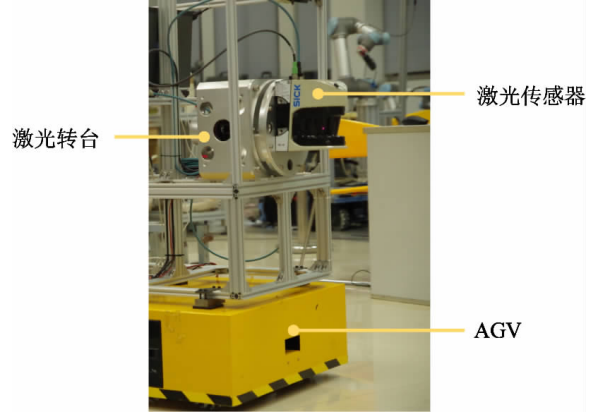


图 6 3D 环境重建实验平台

所有测试程序都在一台 CPU 为 Intel Core-i7 3612QM,内存为 8G 的笔记本电脑上运行,与本文算法进行对比的正态分布变换 (NDT) 算法和迭代最近点 (ICP) 算法来自于 PCL 库<sup>[22]</sup>。

本文在一间长宽高约为 12m × 10m × 6m 的实验室内不同位置采集了 22 组点云。拼接前的点云如图 7 所示。

使用本文提出的方法将图 7 所示的 22 组点云依次进行配准,各次配准所用的时间如图 8 所示,由于在不同位置采集到的点云的数量不同,因此在不同位置配准所用的时间不同,但 22 组实验的配准耗时都在 0.2s 以内。不同灰度代表了配准的不同阶段消耗的时间,可以看到,点云配准过程的主要消耗的时间是平面提取的时间,由于直接使用点云平面不再需要对大量点云进行迭代运算,所以运算速度非常快。配准的结果则如图 9 所示。



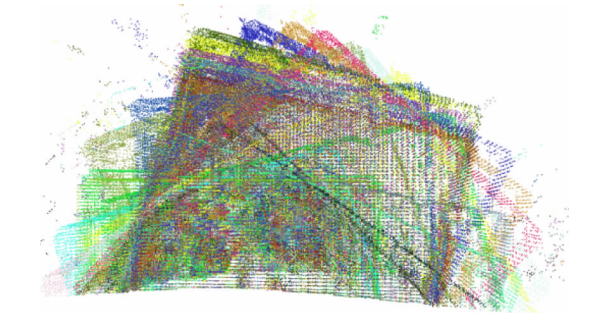


图 7 原始点云数据

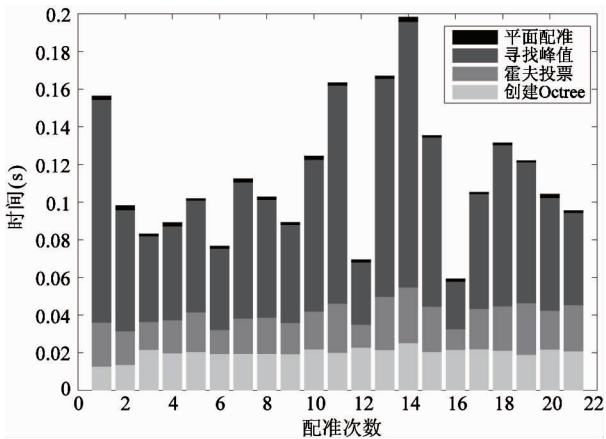


图 8 对 22 帧点云依次配准所用时间

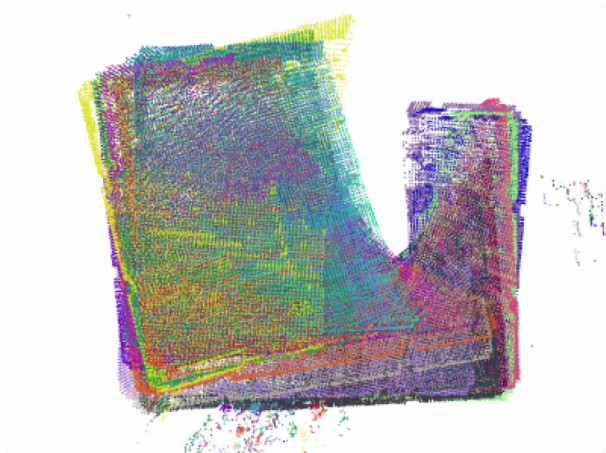


图 9 本文的配准算法配准结果

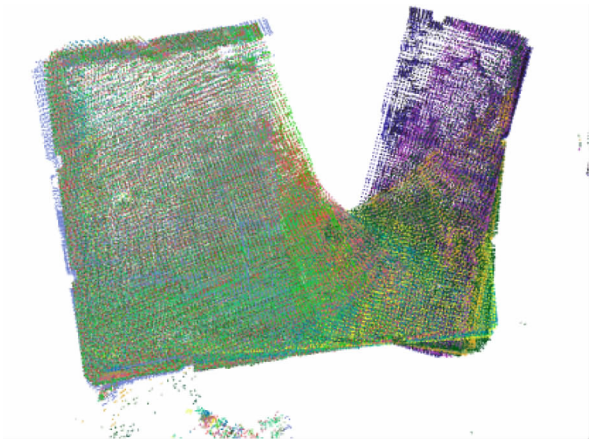


图 10 ICP 配准结果

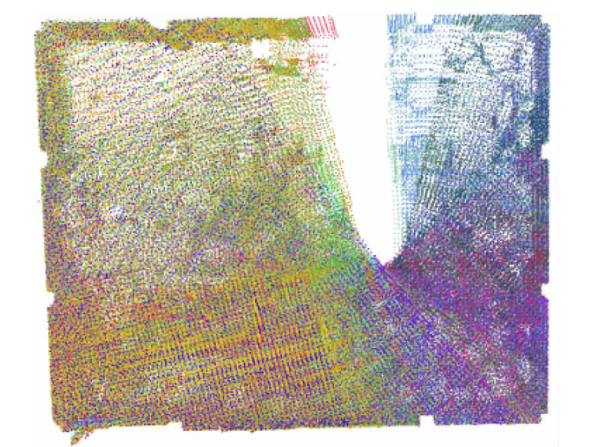


图 11 NDT 配准结果

但是本文提出的算法在配准时间上具有明显的优势,表 1 所示为三种方法配准这 22 帧点云消耗的平均时间,图 12 所示为用这三种方法配准消耗时间的盒式图。由于一帧点云中大约只有几十个平面,因此我们只需要对几十个特征点进行配准,远小于原始点云中的上万个点,因此本文方法的配准速度比 ICP 和 NDT 快了约两个数量级,达到了实时配准的要求。

为了对比,本文也使用了 ICP 和 NDT 对该点云数据进行了配准,配准结果分别如图 10 和图 11 所示。

比较这三个配准结果可以发现,NDT 的配准精度最好,而本文的方法和 ICP 则会有较为明显的累计误差。这主要是由于本算法和 ICP 算法都采用了寻找欧氏距离最近的点作为匹配点,而这将不可避免地陷入局部最优解。

表 1 三种方法的平均消耗时间

| 方法   | 平均消耗时间(s) |
|------|-----------|
| 本文方法 | 0.11      |
| ICP  | 12.50     |
| NDT  | 76.20     |

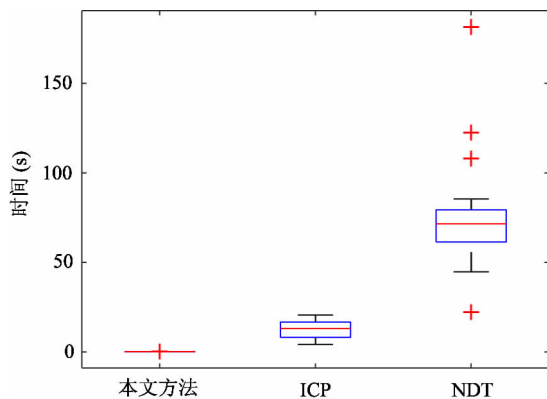


图 12 三种算法消耗时间的盒式图

## 4 结 论

本文提出了一种使用点云平面特征进行快速点云配准的方法。通过在平面的参数空间使用迭代算法进行平面配准,在普通 CPU 上实现了接近实时的点云配准速度。本算法适合于室内服务机器人,救援机器人等对定位实时性要求比较高的场合。或者,本方法也可以作为其他配准方法的一个预处理步骤,即先通过本方法估计出一个点云的变换关系,然后再用 NDT 等其他算法进行配准,可以加快其它算法的配准速度。

## 参考文献

- [1] Besl P J, McKay H D. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992, 14(2): 239-256
- [2] Magnusson M. The three-dimensional normal-distributions transform: an efficient representation for registration, surface analysis, and loop detection: [Ph. D Dissertation]. Örebro: Örebro University, 2009. 55-102
- [3] Izadi S, Kim D, Hilliges O, et al. KinectFusion: Real-time 3D reconstruction and interaction using a moving depth camera. In: *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, Santa Barbara, USA, 2011. 559-568
- [4] 朱笑笑,曹其新,杨扬等. 一种改进的 KinectFusion 三维重构算法. *机器人*, 2014, 36(2): 129-136
- [5] Endres F, Hess J, Sturm J, et al. 3-D mapping with an RGB-D camera. *IEEE Transactions on Robotics*, 2014, 30(1): 177-187

- [6] Kinect for Windows Sensor Components and Specifications: Depth Space Range. [https://msdn.microsoft.com/en-us/library/hh973078.aspx#Depth\\_Ranges](https://msdn.microsoft.com/en-us/library/hh973078.aspx#Depth_Ranges); Microsoft Corporation, 2012
- [7] Kinect hardware key features and benefits. <https://developer.microsoft.com/en-us/windows/kinect/hardware>; Microsoft Corporation, 2015
- [8] ASUS Xtion Pro Specifications. [https://www.asus.com/3D-Sensor/Xtion\\_PRO/specifications/](https://www.asus.com/3D-Sensor/Xtion_PRO/specifications/); ASUS Corporation, 2016
- [9] Intel® RealSense™ Data Ranges. <https://software.intel.com/en-us/articles/intel-realsense-data-ranges>; Intel Corporation, 2016
- [10] Weingarten J, Siegwart R. 3D SLAM using planar segments. In: *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, 2006. 3062-3067
- [11] Pathak K, Birk A, Vaskevicius N, et al. Fast registration based on noisy planes with unknown correspondences for 3-D mapping. *IEEE Transactions on Robotics*, 2010, 26(3): 424-441
- [12] Rusu R B. Semantic 3D object maps for everyday manipulation in human living environments: [Ph. D dissertation]. Munich: Technical University of Munich, 2010. 23-28
- [13] Taguchi Y, Jian Y D, Ramalingam S, et al. Point-plane SLAM for hand-held 3D sensors. In: *Proceedings of the 2013 IEEE International Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, 2013. 5182-5189
- [14] 肖博文. 基于平面提取的三维点云点配准算法研究: [硕士学位论文]. 北京:中国科学院大学, 2016. 19-43
- [15] Fischler M A, Bolles R C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun ACM*, 1981, 24(6): 381-395
- [16] Fernandes L A F, Oliveira M M. Real-time line detection through an improved Hough transform voting scheme. *Pattern Recognition*, 2008, 41(1): 299-314
- [17] Trevor A J B, Gedikli S, Rusu R B, et al. Efficient organized point cloud segmentation with connected components. In: *Proceedings of the Semantic Perception Mapping and Exploration (SPME)*, Karlsruhe, Germany,

2013

[18] Limberger F A, Oliveira M M. Real-time detection of planar regions in unorganized point clouds. *Pattern Recognition*, 2015, 48(6): 2043-2053

[19] Dunteman G H. Principal Components Analysis. Los Angeles: SAGE Publishing, 1989. 23-35

[20] Borrmann D, Elseberg J, Lingemann K, et al. The 3D Hough transform for plane detection in point clouds: a review and a new accumulator design. *3D Research*, 2011,

2(2): 3

[21] Papadimitriou P. Parallel Solution of SVD-related Problems, with Applications:[Ph. D Dissertation]. Manchester: University of Manchester, 1993. 9-10

[22] Rusu R B, Cousins S. 3D is here: point cloud library (PCL). In: Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 2011. 1-4

## A method for real-time 3D point cloud registration in a parameter space of planes

Wang Liyu, Cao Qixin, Wang Wenshan  
(State Key Laboratory of Mechanical System and Vibration, Research Institute of Robotics,  
Shanghai Jiaotong University, Shanghai 200240)

### Abstract

For 3D localization and modelling of an indoor environment with a multi-plane structure, a new method for real-time 3D point cloud registration in a planar parameter space was proposed. The method uses an improved 3D hough transform to quickly extract the plane features in point clouds, then, searches the closest planes in the parameter space of the planes by using an iterative method, and finally, estimates the translation and rotation of the point clouds by registration of the corresponding planes. A comparison experiment performed in a laboratory environment showed that the registration accuracy of the proposed method reached the traditional iterative closest point (ICP) method, and the registration speed was greatly improved. The new method can achieve the real-time registration of point clouds when using an ordinary laptop CPU.

**Key words:** point cloud registration, plane extraction, Hough transform, 3D map reconstruction, iterative closest point (ICP)