3D Point Cloud Registration Based on Planar Surfaces

Junhao Xiao and Benjamin Adler
Department of Computer Science
University of Hamburg
Hamburg, Germany
{xiao, adler}@informatik.uni-hamburg.de

Houxiang Zhang
Department of Technology and Nautical Sciences
Aalesund University College (AAUC)
Aalesund, Norway
hozh@hials.no

Abstract—this paper focuses on fast 3D point cloud registration in cluttered urban environments. There are three main steps in the pipeline: Firstly a fast region grow planar segmentation algorithm is employed to extract the planar surfaces. Then the area of each planar patch is calculated using the image-like structure of organized point cloud. In the last step, the registration is defined as a correlation problem, a novel search algorithm which combines heuristic search with pruning using geometry consistency is utilized to find the global optimal solution in a subset of \( SO(3) \times R^3 \), and the transformation is refined using weighted least squares after finding the solution. Since all possible transformations are traversed, no prior pose estimation from other sensors such as odometry or IMU is needed, making it robust and can deal with big rotations.

Index Terms—Point cloud registration, planar segmentation, segment area calculation, SLAM

I. INTRODUCTION

Simultaneous localization and mapping (SLAM) is indispensable for field service robots, which is needed for many tasks. It has been researched intensively in 2D space in the last decade, two parts of a comprehensive overview can be found in [1], [2]. Among the methods, the particle-filter is proven to be accurate, fast and robust. However, it cannot be adapted into 3D space due to high computational complexity: the robot’s pose to be tracked in 2D and 3D are three and six dimension, respectively.

Different sensors have been employed in 3D SLAM systems, such as monocular cameras [3], stereo vision systems [4], laser range finders (LRF) [5], [6], time-of-flight (ToF) cameras [7] and the recently developed RGB-D cameras [8]. Since other sensors are sensitive to the illumination condition, LRF is state-of-the-art sensor for metric mapping in outdoor environments, and point cloud registration serves as the base of LRF SLAM systems. After registration, scans sampled at different locations are aligned into a global coordinate system, thus constructing a map.

Iterative Closest Point (ICP) [9] is the golden standard method for registration, however it needs a prior estimate of the robot’s position from other sensors such as odometry and initial inertial measurement unit (IMU). However, IMU is relative expensive and odometry cannot provide reliable pose estimation in rough terrain environments, which challenges ICP in outdoor environments. The recently proposed Normal Distribution Transformations (NDT) [6] has the same problem.

We present a novel registration algorithm which is defined as a to maximize the spherical correlation on \( S^2 \), the maximum is found by a novel search algorithm, which combines heuristic search with pruning using geometry consistency and boundaries of the rotation and translation, employing big planar patches’ attributes. Since all the possible transformations are traversed, it does not need prior pose estimation from other sensors. The planar patches are extracted using a point-based region growing algorithm. The surfaces are assigned two attributes which are employed in the search algorithm. One is the Hessian form plane equation which is fitted to each planar surface during the region growing phase, the other one is the area which is calculated as the sum area of local small triangles and quadrilaterals. Then only the planar patches with big area are considered for solution searching, the transformation is refined by weighted least squares using the found solution in the end.

This paper is organized as follows, the related work on plane-based registration will be given in Section II, followed by detailing the proposed registration pipeline in Section III. Section IV presents the experiments and results and Section V draws the conclusion and future work, respectively.

II. RELATED WORK

Various kinds of objects with planar surfaces can be found in both indoor and urban environments, such as floors, doors, walls, ceilings and roads. If the surfaces can be segmented as polygons, they can have numbers of applications, such as compressive representation, 3D building model reconstruction [10], semantic map construction [11], object recognition [12] and especially SLAM [13]–[15]. Moreover, 3D Plane SLAM have been proven to be faster and more reliable than ICP and NDT.

Researchers have proposed several algorithms on 3D Plane SLAM. From the literature finding corresponding (infinite) plane pairs in successive scans is the core problem in the 3D plane SLAM systems. Weingarten and Siegwart [13]
represented the planar patches in a stochastic map and employ an extended Kalman Filter (EKF) to track the pose of the robot. Data association is done by minimizing the Mahalanobis distance between features which are derived from planar patches. In Viejo and Cazorla [14], an ICP-like method is used to address the SLAM problem based on implicit small planar-patch correspondences. The pose estimation from odometry is needed in these two above approaches which restrict their application in outdoor environments. Harati and Siegwart proposed a light SLAM algorithm which takes advantage of orthogonality in the engineered structures of man-made environments in [16], the so called right angle helps to determine plane correspondences, which is limited to indoor environments. A more recent work has been reported in [15], where the plane parameter uncertainties of each segment are calculated during weighted least-squares plane fitting. Armed with this attribute, they proposed the minimally uncertain maximal consensus (MUMC) algorithm, which is capable of determining the plane correspondences by maximizing the geometric consistency while minimizing the resulting uncertainty.

Notably, none of the above approaches have exploited a plane’s area to help identify correspondence between multiple point clouds. Different from them, Makadia et al. proposed to construct a so-called constellation image for each point cloud based on the planar patches’ normal and area, which is used for spherical correlation in [17]. Although the $SO(3)$ Fourier transform (SOFT) is utilized to reduce the complexity of searching solution among all rotations in the rotation group $SO(3)$, the complexity is still high without heuristic and pruning technique. Moreover, the constellation image is usually sparse which means the solution can only appear in a subspace of the rotation group. Another defect of this approach is that only a coarse transformation estimate is given which needs to be refined by ICP.

III. PLANE BASED REGISTRATION

The basic steps of our registration algorithm are depicted in Fig.1 and described in more detail in the remaining parts of this section.

A. Planar segmentation

A point-based region growing (PBRG) is employed for planar segmentation, which is proposed in the authors’ previous work. We give a brief overview here, please refer to [18] for detail. The algorithm proceeds as follows: a local plane is fitted to each point and its nearest pixel-neighbors. Afterwards, the local plane with the smallest fitting mean square error (MSE) among all unidentified points is chosen as a new seed $GR$. Then $GR$ is extended by investigating its neighbors which are within distance $\delta$. Suppose the considering point is $p_c$, it will be added to $GR$ if: First, the distance from $p_c$ to the optimal plane of $GR \cup p_c$ is less than $\gamma$; Second, the MSE of the points in $GR \cup p_c$ to the optimal plane is less than $\epsilon$. If $p_c$ is put into $GR$, its nearest pixel-neighbors will be marked as $GR$’s neighbors which are investigated later. The growth will continue until no more points can be added. Finally, it is assigned to be a new planar patch and added to the segments set $\mathbb{R}$ if it contains more than $\theta$ points. Otherwise, it is added to the uncertain points set $\mathbb{R}'$.

The algorithm is good at speed, however it can only be applied to organized point clouds, because the pixel neighborhood information has been employed for efficient nearest-neighbors operations. After the planar segmentation, an infinite plane is fitted to each segment using least squares. Its equation is assigned to the corresponding segment in the Hessian form:

$$\hat{n} \cdot p = d,$$

where $\|\hat{n}\| = 1$ is the normalized surface normal and $d$ ($d > 0$ for consistency) is the distance from it to the coordinate origin.

B. Segment area calculation

The area covered by coplanar points in 3D space can be calculated if their shape (usually as polygon) can be determined. Available algorithms figuring out the covered shape of coplanar points can be found in the computational geometry literature [19], such as the convex hull and alpha shape approach. Results of them are convex polygons or general polygons (which can have holes), respectively. The convex hull algorithm is designed for convex subsets in $\mathbb{R}^2$, which cannot always be satisfied in real-world environments, restricting its application for segment area computation. The 2D alpha shape ($\alpha$-shape) algorithm can be used for shape

![Fig. 1. Framework of the proposed registration approach, there are three main parts: 1. fast region growing planar segmentation, 2. planar segment area calculation based on local triangles and quadrilaterals, 3. solution searching using heuristic and pruning.](image-url)
reconstruction from a dense unorganized set of points and results in a linear approximation of the original shape. One challenge of it is to determine an optimum $\alpha$ value for each segment. Although strategies for searching optimum $\alpha$ values do exist, they rarely provide satisfying results. As a compromise, finding the ideal $\alpha$ is usually an interactive process, which limits its usage in robotic systems.

Since traditional algorithms from computational geometry proved unfit to process data from real-world sensors, we proposed a novel approach inspired by the surface integral. The idea is to use the sum of small local surfaces to approximate the segment’s area. Now it is reformulated into two sub-problems: one is to find local faces and the other one is to calculate the area of local faces, here local faces stand for triangles and quadrilaterals.

We address the first sub-problem using the image-like structure of organized point cloud. It is already known which points are coplanar and on the same surface after planar segmentation, thus finding local triangles and rectangles is straightforward. The triangles remain as triangles when projected from pixel to 3D space, while the rectangles become general quadrilaterals.

The second sub-problem can be stated as follows: given a planar polygon $\Omega$ (triangle or quadrilateral) in 3D, with vertices $v_0, v_1, \ldots, v_M$ ($v_M = v_0$, $M = 3$ and 4 for triangle and quadrilateral respectively) in counter-clockwise order, the aim is to calculate its area. After some computational geometric derivation, it yields

$$s(\Omega) = \frac{1}{2} \sum_{i=0}^{M-1} v_i \times v_{i+1}$$  \hspace{1cm} (2)

where $\hat{n}$ is the surface normal of $\Omega$.

\textbf{C. Solution searching and transformation calculation}

The problem can be stated as: Given two successive overlapping point clouds $C_m$ and $C_d$, which are sampled by the robot at location $O_m$ and $O_d$ in the robot-frames $F_m$ and $F_d$ respectively, the aim is to find the 3D rotation $m \ R$ and translation $m \ t$ which satisfies the relation

$$m \ p = m \ R \ d \ p + m \ t$$  \hspace{1cm} (3)

for a point observed in both point clouds with coordinates $m \ p$ and $d \ p$, respectively.

After planar segmentation and segment area calculation, each point cloud has been represented as indexed planar patches using a set of triplets

$$k \ P = \{ k \ P_1, k \ n_i, k \ d_i, k \ s_i \}, i = 1, 2, \ldots, N_k, k = m, d$$  \hspace{1cm} (4)

The desired $m \ R$ and $m \ t$ will maximize the following correlation

$$\max_{m \ R, m \ t} \sum_{i=1}^{N_m} \sum_{j=1}^{N_d} m \ s_i \ d_j s_j \sigma_{ij},$$  \hspace{1cm} (5)

where $\sigma_{ij}$ is geometric consistency and the robot’s kinematic model.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{algorithm_flowchart.png}
\caption{Flow chart of the proposed transformation search algorithm, all possible rotations $R$ and translations $t$ are traversed heuristically by two and three unparallel correspondence plane pairs, and the search path is pruned by geometric consistency and the robot’s kinematic model.}
\end{figure}
\( \sigma_{ij} \) is defined as

\[
\sigma_{ij} = \begin{cases} 
1 & \text{if } mP_i \leftrightarrow dP_j \\
0 & \text{otherwise}
\end{cases}
\] (6)

where \( mP_i \leftrightarrow dP_j \) means \( mP_i \) and \( dP_j \) are the same planar surface which is observed in both \( C_m \) and \( C_d \). Now the problem becomes how to determine the plane correspondences, this is solved by the following novel heuristic search algorithm which has been depicted in Fig.2. Firstly we enumerate all potential corresponding planar pairs which have similar area, \( mP_i \) and \( dP_j \) are marked as a potential correspondence if

\[
\left( \frac{|m s_i - d s_j|}{\max(m s_i, d s_j)} \right) < \lambda,
\] (7)

planar patches with area smaller than \( \tau \) are ignored in order to avoiding noisy correspondences and guarantee the registration speed. All founded plane pairs are put into a list \( \mathcal{L} = \{\mathcal{L}_i \mid \langle mP_{i1}, dP_{i2}\rangle, i = 1, \ldots N_k, mP_{i1} \in mP, dP_{i2} \in dP\} \). (8)

As it is known, the rotation matrix and translation vector can be computed by two and three unparallel correspondence planes, respectively. Two potential correspondence pairs are treated as unparallel if they fulfill the conditions illustrated at the bottom of Fig. 2. In other words, two planes in the map cloud has similar angle as the other two planes in the data cloud, and the angle is not 0 or \( \pi \). The a rotation matrix is calculated by the fixed two plane pairs, which is evaluated by the robot’s kinematic model in order to restrict the rotation angle to be realistic. For this purpose, we decompose the rotation into two successive rotations in the form of axis-angle: \( \langle \hat{h}, \alpha \rangle \) and \( \langle \hat{z}, \beta \rangle \), where \( \hat{h} \cdot \hat{z} = 0 \), and \( \hat{z} \) is along the the z-axis. According to the robot’s locomotion ability, \( \alpha \) should be smaller than a angle threshold \( \chi \) and there is no limitation for \( \beta \). If the rotation is realistic, we will try to find another plane pair \( \mathcal{L}_k \) which satisfies the following conditions: \( \mathcal{L}_k \) is unparallel to both \( \mathcal{L}_i \) and \( \mathcal{L}_j \) (formulated at the bottom of Fig. 2) and agrees with this rotation, i.e.,

\[
m\hat{n}_{k1} \cdot (R \ast d\hat{n}_{k2}) \approx 1.
\] (9)

If \( \mathcal{L}_k \) is found, a translation \( t \) is calculated using the three plane pairs. The translation is then also assessed using the robot’s kinematic model, i.e., the angle

\[
\cos \left( \frac{t \cdot \hat{z}}{\sqrt{(t \cdot x)^2 + (t \cdot y)^2}} \right)
\] (10)

should be smaller than \( \chi \). If \( t \) qualifies the test, other correspondence plane pairs which are consensus with \( \langle R, t \rangle \) can be found by geometric consistency assessments, this test should be conducted on all other items in \( \mathcal{L} \) except the three fixed pairs. Suppose \( \mathcal{L}_c \) is investigated, it agrees with \( \langle R, t \rangle \) if

\[
\begin{align*}
&m\hat{n}_{k1} \cdot (R \ast m\hat{n}_{k2}) > \xi \\
&m\hat{n}_{k1} \cdot t + m\hat{d}_{k2} - m\hat{d}_{k1} < \eta
\end{align*}
\] (11)

where \( \xi \) and \( \eta \) are two scalar thresholds. Afterwards, the correlation in (5) will be computed for the found consensus correspondences \( \Phi \) if it has more than \( \Theta \) correspondences. The solution with biggest correlation value will be selected in the end.

Since the searched rotation and translation are calculated by only two and three plane pairs, they are refined using least squares using all consensus correspondences and the least squares is solved by singular value decomposition (SVD).

IV. EXPERIMENT AND RESULTS

A. Outdoor platform and datasets

The data are carried out by a customized outdoor mobile robot with a virtual 3D LRF. One PTU-D48E unit is employed to yaw the 2D LRF, constructing the 3D scanner. The PTU has continuous-pan-motion thanks to its build-in slip ring. One Hokuyo UTM-30lx LRF is installed horizontally on the top bracket of D48E, with its “Sensor Front” point up as shown in Fig.3. The 2D LRF’s sensor range and field of view (FoV) are 30m and 270 deg respectively. During each scan, all 2D scan slices joint at one point, i.e., the “sensor front”.

We construct a 3D Organized Point Cloud for each 3D scan which maintain the laser beam adjacent information during scanning, the advantages of such point cloud is that the relationship between adjacent points is known, making nearest-neighbor operations much more time-efficient, at the price of more storage space is needed for invalid points. The organized point cloud is constructed as follows: treat each half 2D scan slice (from the sensor front to the begin step or the end step) as a row, and the adjacent half slices are stored as adjacent rows in the resulted image-like structure.

The dataset is gathered in the Department of Informatics, University of Hamburg. The pan resolution is set to 0.5 degree and the 2D scan resolution is 0.25 degree, thus construct organized point clouds with 720 rows and 540 columns, in total 388,800 points. Besides the depth information, the
intensity information has also been provided in the dataset, one typical point cloud together with its correspondence depth- and intensity- image have been illustrated in Fig.4. Such a dataset can be used not only for evaluating SLAM systems but also for intensity and depth information fusion.

We choose to use the “bridge” area between two buildings to evaluate the proposed registration pipeline, which is drawn in Fig.5. The robot is controlled remotely, in total 8 point clouds are gathered and big orientation changes have been made between successive scans, see Fig.6 as an example. The robot has travelled about 20 meters during the experiment, i.e., approximate 3 meters between each scan, which is relative big comparing to the literature thus challenges the registration algorithm.

B. Implementation

The approach has been implemented using c++, the point cloud library (pcl) [20] is employed for data handling and visualization. Linear algebra is crucial in the approach, especially for calculating the eigenvalues and eigenvectors of a square matrix, since it has to be performed whenever one point is investigated in planar segmentation. Therefore the Eigen library [21] was employed in the implementation. The experiments have been run on an Intel Core 2 Duo 2.53GHz, 2GB RAM under Ubuntu 11.10. Parameters has been tuned for planar segmentation and solution searching. The parameters used in the experiments is shown in Tab.I.

<table>
<thead>
<tr>
<th>Parameters for the 3D LRF</th>
</tr>
</thead>
<tbody>
<tr>
<td>planar segmentation</td>
</tr>
<tr>
<td>transform</td>
</tr>
<tr>
<td>search</td>
</tr>
</tbody>
</table>

C. Results

For planar segmentation and area calculation, the result of Fig.4 is shown in Fig.7, planar patches are colored randomly and the corresponding area is drawn at their geometric center. The registration result of the 8 point clouds is illustration in Fig.8. Since the ground truth has not been built, the result is inspected by eyes and no big error was found. It verifies that the proposed approach is able to deal with big rotation and translation without prior pose estimation.
for better visualization.

Fig. 7. Planar segmentation and segment area calculation result of Fig.4, the planar patches have been colored randomly and the correspondence area is drawn at their geometric center.

Fig. 8. The registration result of 8 point clouds, pseudo color is employed for better visualization.

V. CONCLUSION AND FUTURE WORK

We proposed a 3D point cloud registration approach which is composed of three parts: planar segmentation, planar segment area calculation and transformation searching. Since the algorithm searches all possible transformations (rotation and translation) globally, no prior pose estimation is needed, making it suitable for outdoor environments. The main disadvantage is the requirements of planar surfaces, however, this is not a problem for urban environments.

In the future, we will benchmark the algorithm using different sensor setups and environments on accuracy, robustness and speed, then we will try to build a SLAM system based on the registration algorithm.

REFERENCES


