A Plane-based Approach for Indoor Point Clouds Registration

Ketty Favre, Muriel Pressigout, Eric Marchand, Luce Morin

HAL Id: hal-03108891
https://hal.archives-ouvertes.fr/hal-03108891

Submitted on 13 Jan 2021

A Plane-based Approach for Indoor Point Clouds Registration

Ketty Favre*   Muriel Pressigout†   Eric Marchand‡   Luce Morin†

*Univ Rennes, CNRS, IETR - UMR 6164, Rennes, France.
†Univ Rennes, INSA Rennes, CNRS, IETR - UMR 6164, Rennes, France.
‡Univ Rennes, Inria, CNRS, IRISA, Rennes, France.
Email: ketty.favre@univ-rennes1.fr

Abstract—Iterative Closest Point (ICP) is one of the mostly used algorithms for 3D point clouds registration. This classical approach can be impacted by the large number of points contained in a point cloud. Planar structures, which are less numerous than points, can be used in well-structured man-made environment. In this paper we propose a registration method inspired by the ICP algorithm in a plane-based registration approach for indoor environments. This method is based solely on data acquired with a LiDAR sensor.

A new metric based on plane characteristics is introduced to find the best plane correspondences. The optimal transformation is estimated through a two-step minimization approach, successively performing robust plane-to-plane minimization and non-linear robust point-to-plane registration.

Experiments on the Autonomous Systems Lab (ASL) dataset show that the proposed method enables to successfully register 100% of the scans from the three indoor sequences. Experiments also show that the proposed method is more robust in large motion scenarios than other state-of-the-art algorithms.

I. INTRODUCTION

In robotics, registration of 3D point sets is a key issue in localization applications. The trend for autonomous vehicles makes it a widely searched field. Nowadays, 3D LiDARs are becoming cheaper and more frequently used. They have proven their efficiency in localization applications. The raw data generated by a 3D LiDAR are 3D point clouds, meaning a set of 3D points representing the coordinates of the physical point hit by the laser in the sensor reference frame.

One of the most popular approaches in robotics to register 3D point clouds is the well-known Iterative Closest Point (ICP) algorithm [1]. It allows to compute the rigid transformation (rotation and translation) that links a source and a target 3D point cloud. To do so, each point from the source point cloud is paired with its closest point in the target one. Then the 3D rigid transformation that minimizes the distance between paired points is estimated. This is achieved within an iterative scheme until the residual error has reached the desired threshold. A survey presenting ICP variants are given in [2] and [3].

This introduction provides a short synthesis of the ICP variants focused on the different distances that can be used in order to estimate the 3D rigid transformation that registers

two point clouds. The point-to-point distance is first introduced in [1]. Then, the point-to-plane distance is presented in [4], proven to be more robust and faster to converge than the point-to-point one. A linear resolution of this minimization can be found in [5] using the small angle approximation to solve the optimization problem. In [6], the problem is kept non-linear and solved using a Levenberg-Marquardt approach integrating robust estimators. In [7], Normal Distribution Transform (NDT) takes into account local surface structures around each point and does not match individual points unlike common ICP variants. In Generalized-ICP (G-ICP) [8], the local neighborhood of points is used in order to assimilate this structure to small planar patches. As in point-to-plane approaches, the local normals of the target point cloud are taken into account but also the ones of the source point cloud. It can be assimilated to plane-to-plane registration.

In [9] and [10], the approaches exploit the planar surfaces of man-made environments with a plane-to-plane distance. Those approaches are interesting to use in indoor environments when enough planes are available. However, segmenting planes can be time consuming. In [11] and [12], the 3D data of the sensor are used as range images. The neighborhood relation of the pixels is then used to segment the planes in a region growing scheme. It also includes a polygonalization of the planes in surface models. A region growing process based on smoothness is introduced in [13]. In [14], [15] and [9], RANSAC approaches are used in order to fit points to planar patches.

Matching planes after the segmentation is another challenging task. In [10], the data rate acquisition is supposed very high, which leads to low relative translation between scans. Thus, planes with the projections of the origin of the sensor close to each other and almost parallel are matched. In [16], a plane/line descriptor is proposed to establish structure correspondences. Attributes of the planes and the constraints between them are used in [17].

Algorithms such as G-ICP [8], point-to-plane ICP [5] and NDT [7] while being efficient for fine registration are sensitive to large motion and usually need a good initialization of the rigid transformation in order to converge. Moreover G-ICP can be slow to perform registration as is shown further in this article. In this paper, we exhibit a framework designed for
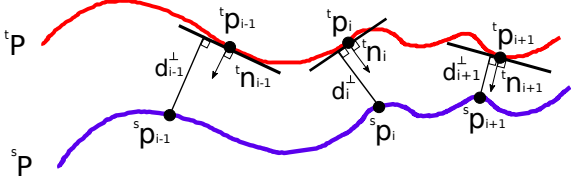
Fig. 1. Point-to-plane distance $d_i^\perp$ as described in [5]. In red the surface on which the target point cloud lies and the surface normals related to its points. In blue the surface where lies the source point cloud to register.

plane-to-plane registration in indoor environment. This method is denoted New Accurate Plane-based ICP (NAP-ICP). The proposed algorithm is robust to large motion, thus it is less sensitive to initialization than other evaluated state-of-the-art algorithms. The main contributions of this article are:

- an efficient score metric for finding best plane correspondences;
- a two-step minimization method from coarse to fine registration based on plane features;
- an algorithm performing fast and accurate registration in challenging datasets;
- a method robust to large motion or inaccurate initialization.

First the variants of the distance to minimize are presented. After the methodology of the proposed method is described with a section related to each step of the algorithm. Then the experiments and their results are presented. Finally a conclusion and perspectives are given.

II. ICP VARIANTS

In the following sections, source and target points will be respectively denoted $^s\mathbf{P} = \{^s\mathbf{p}_i\}_{i=1}^{N}$ and $^t\mathbf{P} = \{^t\mathbf{p}_j\}_{j=1}^{M}$. The target point cloud is fixed. The goal is to find the rigid transformation $^t\mathbf{T}_s$ that best fits the source to the target. This transformation is defined as follows:

$$^t\mathbf{T}_s = \begin{bmatrix} ^t\mathbf{R}_s & ^t\mathbf{t}_s \\ \mathbf{0}_{3\times1} & 1 \end{bmatrix} \tag{1}$$

with $^t\mathbf{R}_s$ and $^t\mathbf{t}_s$ respectively a 3×3 rotation matrix and a 3×1 translation vector.

Each point $^s\mathbf{p}_i$ of the source is matched with its closest point $^t\mathbf{p}_i$ in the target. Then the rigid transformation minimizing a distance metric is estimated and these two steps are iterated until a threshold is reached. In the original ICP algorithm [1] the distance metric to be minimized is the Euclidean point-to-point distance.

As corresponding points from one scan to another may not be exactly identical but may lie on the same surface, it is better to choose, as in [4], to minimize the point-to-plane distance (Fig. 1), defined by:

$$d_i^\perp = \|^t\mathbf{n}_i^\top \cdot (^t\mathbf{T}_s\,^s\mathbf{p}_i - {}^t\mathbf{p}_i)\|^2 \tag{2}$$

with $^t\mathbf{n}_i$ the surface normal computed from $^t\mathbf{p}_i$ neighborhood.

In order to reduce the number of input data in the optimization step, the plane-to-plane distance is used. A plane $\Pi(\rho, \mathbf{n})$ is given by the equation $\mathbf{n}^\top \mathbf{p} = \rho$, where ρ is the distance from the origin of the sensor in the direction of the unit plane normal $\mathbf{n}$. The distance between two corresponding planes ([9], [10]), with $^s\Pi_i(^s\rho_i, {}^s\mathbf{n}_i)$ the source plane and $^t\Pi_i(^t\rho_i, {}^t\mathbf{n}_i)$ the target one, is given as follows:

$$\mathbf{d}_i^\Pi = \begin{pmatrix} ^t\mathbf{R}_s\,^s\mathbf{n}_i - {}^t\mathbf{n}_i \\ [^t\mathbf{R}_s\,^s\mathbf{n}_i]^\top\,^t\mathbf{t}_s + {}^s\rho_i - {}^t\rho_i \end{pmatrix} \tag{3}$$

where $^s\mathbf{n}_i$ and $^t\mathbf{n}_i$ are the normal to $^s\Pi_i$ and $^t\Pi_i$ respectively and $^s\rho_i$ and $^t\rho_i$ their respective distance to the origin of the sensor in the target frame.

III. METHODOLOGY

This section describes each step of the proposed NAP-ICP algorithm. The framework is given in Fig. 2. Similarly to the classical ICP algorithm the method iteratively performs the matching step and the minimization step. However in the proposed method the first features to be matched are planes. Once matched, the rigid transformation minimizing the plane-to-plane distance is estimated. After the plane-to-plane registration is performed, an additional point-to-plane registration is done. An example of registration using the proposed method is given in Fig. 3.
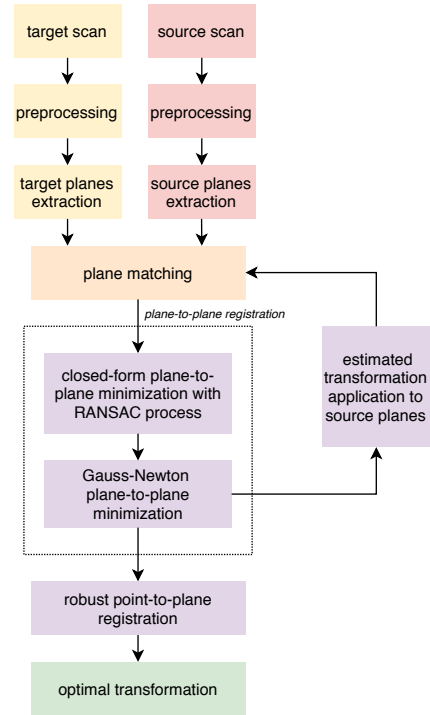

Fig. 2. The proposed NAP-ICP overview.

Each step of the framework is described further in this article:

- preprocessing of the 3D point clouds is briefly discussed in section III-A;
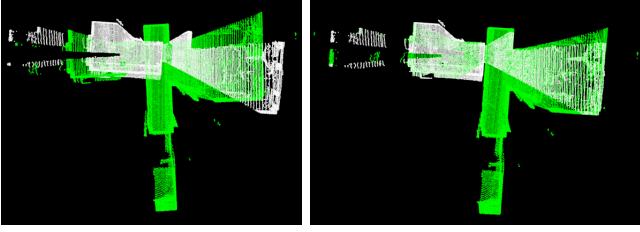- plane extraction is described in section III-B;

Fig. 3. Example of the registration between two point clouds (scans 3 and 4 from *Apartment* sequence from ASL dataset [18]). The overlap between scans is small, yet the proposed method succeeds in registering the two point clouds accurately. In white the target point cloud - In green the source point cloud. *Left:* before registration - *Right:* after registration.

- the score metric for finding best plane correspondences is detailed in section III-C;
- robust plane-to-plane registration is described in section III-D;
- the additional point-to-plane minimization leading to finer registration is detailed in section III-E.

A. Preprocessing

In order to perform plane-to-plane registration as fast and accurately as possible preprocessing is sometimes needed. To speed up computation time, target and source scans are subsampled using a voxel grid of a given resolution. Also, to avoid points from the acquisition system or operators to be part of the point cloud, all points closer than 50cm to the sensor are discarded.

B. Plane extraction

The first step in a plane-based registration (besides preprocessing) is to extract planar structures. In the presented results, planes are extracted using a region growing segmentation based on [13], using the Point Cloud Library (PCL) [19]. In this approach, the points in a neighborhood with a small angle difference between normals are considered to be on the same smooth surface and are gathered in a cluster. Each cluster represents a plane. The normals are estimated by performing a Principal Component Analysis (PCA) on the neighborhood of the concerned points [19]. An example of the obtained plane segmentation is given in Fig. 4.
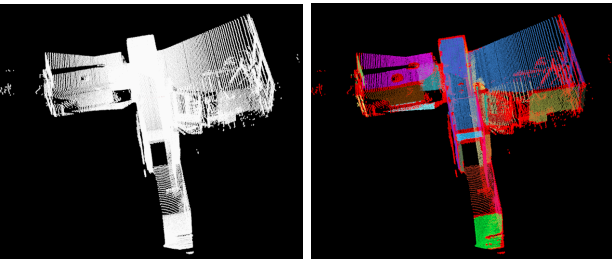

Fig. 4. Plane segmentation with region growing approach. *Left:* input point cloud (from ASL dataset [18]) - *Right:* plane extraction result using region growing [13]. Each extracted plane is in a different color. Red points are outliers.

C. Plane matching

Once the planes are segmented, the next step is matching each source plane to the closest one in the target point cloud. For each extracted plane $^s\Pi_i$ in the source, a list of planes in the target that are potential matches for the source plane is made, called target candidates. Each target candidate $^t\Pi_j$ is given a score within the range $[0;1]$. It is computed from the following features:

- the distance between the projections of the origin on source plane and target plane d_o, expected to be close to 0:

$$d_o = \|{}^s\rho_i\,{}^s\mathbf{n}_i - {}^t\rho_j\,{}^t\mathbf{n}_j\|^2 \tag{4}$$

- the distance between the centroids of source and target planes d_c, expected to be close to 0:

$$d_c = \|{}^s\bar{\mathbf{p}}_i - {}^t\bar{\mathbf{p}}_j\|^2 \tag{5}$$

with $^s\bar{\mathbf{p}}$ and $^t\bar{\mathbf{p}}$ the centroids of $^s\Pi_i$ and $^t\Pi_j$ respectively;

- the area ratio between the planes S_r, expected to be close to 1 as planes are expected to have similar areas:

$$S_r = \frac{min({}^sS_i, {}^tS_j)}{max({}^sS_i, {}^tS_j)} \tag{6}$$

with ^sS_i and ^tS_j the area of source and target planes respectively;

- the dot product of the normals of the planes ϕ_n, expected to be close to 1 as planes are expected to be almost parallel:

$$\phi_n = {}^s\mathbf{n}_i \cdot {}^t\mathbf{n}_j \tag{7}$$

Each feature is normalized between $[0;1]$ and weighted, which leads to a score defined as follows:

$$score = \alpha \cdot \hat{d}_o + \beta \cdot \hat{d}_c + \gamma \cdot (1 - \hat{S}_r) + \delta \cdot (1 - \hat{\phi}_n) \tag{8}$$

with $\hat{\cdot}$ denoting the normalized value, and the weights α, β, γ and δ subject to:

$$\alpha + \beta + \gamma + \delta = 1 \tag{9}$$

Parameters α, β, γ, δ were chosen empirically to fixed values for all experiments such as: $\alpha = 0.35$, $\beta = 0.4$, $\gamma = 0.1$ and $\delta = 0.15$.

A target plane is considered as a valid matching candidate if it respects the following condition:

$$score < t_{score} \tag{10}$$

Some matched planes following the previous condition happen to be too far from each other, hence their centroids are distant. They are discarded with the following condition:

$$d_c > t_{centroid} \tag{11}$$

If the distance d_c between the centroids is bigger than a threshold $t_{centroid}$, the planes are too far from each other to be a valid correspondence.

Also, matched planes are supposed to have similar areas. To avoid matched planes with a notable difference in area, another condition is added:

$$S_r > t_S \qquad (12)$$

If the area ratio S_r is smaller than a threshold t_S, the correspondence is discarded.

The valid pairs of matched planes form the list of correspondences between source and target planes. The resulting list is used for the transformation estimation. As not only the correspondence with the smallest score are kept but all correspondences respecting the previous conditions, the list may contain several occurrences of the same source plane, with different target planes and vice-versa.

D. Plane-to-plane registration

Now that the set of plane correspondences is built, the plane-to-plane distance minimization that estimates the rigid transformation linking source to target planes can be computed.

Closed-form plane-to-plane optimization method: In this section a closed-form minimization of the plane-to-plane distance is presented. The derivation is similar to the one presented in [9] without the point-to-point correspondences.

Corresponding planes are denoted ${}^s\Pi_i({}^s\rho_i, {}^s\mathbf{n}_i)$ and ${}^t\Pi_i({}^t\rho_i, {}^t\mathbf{n}_i)$. Similarly to the point-to-point problem in [20] and [21] the rotation and translation are decoupled.

The rotation estimation is obtained by minimizing:

$$\sum_{i=1}^{N} = \|{}^t\mathbf{R}_s{}^s\mathbf{n}_i - {}^t\mathbf{n}_i\|^2 \qquad (13)$$

First, a 3×3 correlation matrix $\mathbf{H}$ is built such as:

$$\mathbf{H} = \sum_{i=1}^{N} {}^s\mathbf{n}_i{}^t\mathbf{n}_i^\top \qquad (14)$$

Its Singular Value Decomposition (SVD), $\mathbf{H} = \mathbf{UDV}$, is computed. The optimal rotation matrix ${}^t\hat{\mathbf{R}}_s$ is given by:

$${}^t\hat{\mathbf{R}}_s = \mathbf{VU}^\top \qquad (15)$$

To compute the translation the following equation is minimized:

$$\sum_{i=1}^{N} \|{}^t\mathbf{n}_i^\top{}^t\mathbf{t}_s + {}^s\rho_i - {}^t\rho_i\|^2 \qquad (16)$$

corresponding to solving the linear system $\mathbf{A}^t\mathbf{t}_s = b$ where

$$\mathbf{A} = \begin{pmatrix} {}^t\mathbf{n}_1^\top \\ \vdots \\ {}^t\mathbf{n}_N^\top \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} {}^t\rho_1 - {}^s\rho_1 \\ \vdots \\ {}^t\rho_N - {}^s\rho_N \end{pmatrix}$$

The least-squares solution of this problem is given by ${}^t\hat{\mathbf{t}}_s = \mathbf{A}^+\mathbf{b}$ where $\mathbf{A}^+$ is the pseudo-inverse of $\mathbf{A}$.

RANSAC process: As the previously created correspondence list may contain outliers, it is important to discard them as they can lead to divergence in the minimization step. To do so, a RANSAC process is applied to make the minimization more robust.

Only three non-parallel planes are needed in the source and target respectively. Each sample of the RANSAC algorithm is selected respecting this condition.

Gauss-Newton plane-to-plane minimization: The plane correspondences identified as inliers by the RANSAC process are given as input of the Gauss-Newton approach.

This method requires a minimal representation of the transformation to be estimated ${}^t\mathbf{T}_s$. Such a representation is defined by a 6 dimensional vector denoted $\mathbf{q} = ({}^t\mathbf{t}_s, \theta\mathbf{u})^\top$ where θ and $\mathbf{u}$ are the angle and the axis of the rotation ${}^t\mathbf{R}_s$.

The plane-to-plane error has to be minimized such that:

$$\hat{\mathbf{q}} = \arg\min_{\mathbf{q}} \sum_{i=1}^{N} \|\mathbf{d}_i^\Pi\|^2 \qquad (17)$$

Equation (17) can be solved using the Gauss-Newton algorithm. Solving it consists in minimizing the cost function $E(\mathbf{q}) = \|\mathbf{e}(\mathbf{q})\|$:

$$\mathbf{e}(\mathbf{q}) = \begin{pmatrix} \mathbf{n}(\mathbf{q}) - \mathbf{n} \\ \rho(\mathbf{q}) - \rho \end{pmatrix} \qquad (18)$$

with $\mathbf{n}(\mathbf{q}) = (..., {}^t\mathbf{R}_s{}^s\mathbf{n}_i, ...)$, $\mathbf{n} = (..., {}^t\mathbf{n}_i, ...)$, $\rho(\mathbf{q}) = (..., [{}^t\mathbf{R}_s{}^s\mathbf{n}_i]^\top {}^t\mathbf{t}_s + {}^s\rho_i, ...)$ and $\rho = {}^t\rho_i$ the error vector of the distance between the target point cloud and the source point cloud transformed with the previous estimated transformation.

The first order Taylor approximation gives:

$$\mathbf{e}(\mathbf{q} + \delta\mathbf{q}) \approx \mathbf{e}(\mathbf{q}) + \mathbf{J}(\mathbf{q})\delta\mathbf{q} \qquad (19)$$

where $\mathbf{J}(\mathbf{q})$ is the Jacobian of $\mathbf{e}(\mathbf{q})$ in $\mathbf{q}$.

With the Gauss-Newton method, the solution consists in minimizing $E(\mathbf{q} + \delta\mathbf{q})$ with:

$$E(\mathbf{q} + \delta\mathbf{q}) = \|\mathbf{e}(\mathbf{q} + \delta\mathbf{q})\| \approx \mathbf{e}(\mathbf{q}) + \mathbf{J}(\mathbf{q})\delta\mathbf{q} \qquad (20)$$

The minimization problem can be solved by an iterative least-squares approach which gives:

$$\delta\mathbf{q} = -\lambda\mathbf{J}(\mathbf{q})^+\mathbf{e}(\mathbf{q}) \qquad (21)$$

where λ is a coefficient in $]0, 1]$ and $\mathbf{J}^+$ is the pseudo-inverse of the Jacobian $\mathbf{J}$.

The pose is then updated at each iteration:

$$\mathbf{q}_{k+1} = \mathbf{q}_k \oplus \delta\mathbf{q} = \exp^{\delta\mathbf{q}} \mathbf{q} \qquad (22)$$

where $\oplus$ denotes the composition over se(3) obtained via the exponential map.

Its associated $4N \times 6$ Jacobian matrix $\mathbf{J}$ stacks each Jacobian matrix $\mathbf{J}_i$:

$$\mathbf{J}_i = \begin{pmatrix} \mathbf{0}_{3\times3} & [{}^t\mathbf{R}_s{}^s\mathbf{n}_i]_\times \\ -[{}^t\mathbf{R}_s{}^s\mathbf{n}_i]^\top & 0_{1\times3} \end{pmatrix} \qquad (23)$$

with $[\mathbf{x}]_\times$ the skew matrix of a vector $\mathbf{x}$.

E. Point-to-plane Registration

To ensure an accurate registration, a finer step is added to find the best expected rigid transformation. To do so a point-to-plane registration is added at the end of the process. A robust non-linear minimization of the point-to-plane distance is presented. Each point $^s\mathbf{p}_i$ from $^s\mathbf{P}$ is matched to its closest point $^t\mathbf{p}_i$ in $^t\mathbf{P}$ according to the Euclidean distance. Then the rigid transformation that registers source to target point cloud is computed by minimizing the point-to-plane distance (Eq. 2).

The principle is the same as section III-D with:

$$\hat{\mathbf{q}} = \underset{\mathbf{q}}{\arg\min} \ \sum_{i=1}^{N} \|d_i^{\perp}\|^2 \qquad (24)$$

Its Jacobian $\mathbf{J}_i$ is defined by:

$$\mathbf{J}_i = \begin{pmatrix} -^t\mathbf{n}_i^{\top} & ^t\mathbf{n}_i^{\top} \, [^s\mathbf{p}_i]_{\times} \end{pmatrix} \qquad (25)$$

If outliers are present in the dataset the minimization problem becomes unstable. Using the Gauss-Newton method for the minimization allows to introduce M-estimators, a class of robust functions, in the algorithm to discard outliers [22]. Denoting $\rho(.)$ a robust function, $\hat{\mathbf{q}}$ becomes:

$$\hat{\mathbf{q}} = \underset{\mathbf{q}}{\arg\min} \ \sum_{i=1}^{N} \rho(d_i^{\perp}) \qquad (26)$$

The introduction of M-estimators can be implemented as an Iteratively Re-weighted Least Squares (IRLS) where the error to minimize is defined by:

$$\mathbf{e}_{\rho}(\mathbf{q}) = \mathbf{D}(\mathbf{e}(\mathbf{q})) \qquad (27)$$

and

$$\delta\mathbf{q} = -\lambda(\mathbf{DJ}(\mathbf{q}))^{+}\mathbf{De}(\mathbf{q}) \qquad (28)$$

where $\mathbf{D}$ is a $N \times N$ diagonal matrix containing the weights that reflect the confidence in the data.

IV. EXPERIMENTS

To evaluate the efficiency, accuracy and robustness of the proposed method, experiments in different scenarios are held. The indoor sequences of the ASL Dataset [18] are used. One of the main advantages of this dataset is that each sequence comes with the ground truth poses measured for each scan with millimeter precision. As the proposed NAP-ICP is designed to register point clouds in man-made environments, the outdoor sequences containing various types of surfaces (thus not planar) were discarded. All sequences were recorded using a Hokuyo UTM-30LX.

- *Apartment:* This sequence is designed to evaluate algorithm robustness to outliers coming from dynamic elements (e.g. moved furniture). The sequence was captured moving the sensor on a 2D plane in an apartment. It is a very structured scene (walls, ceiling, floor). The sequence is composed of 44 scans of about 365,000 points.
- *ETH:* This sequence aims to evaluate robustness of registration to repetitive elements. This scene was captured in a long hallway, following a straight path. It is composed

of a wall, a curved ceiling and numerous pillars and arches which are repetitive elements. The sequence is composed of 35 scans of about 191,000 points.
- *Stairs:* This sequence aims to evaluate robustness to rapid variations in scanned volumes. It starts in a long corridor, then a small staircase is crossed and finally the last scan is captured outside of the building. The path in the staircase shows that considering only 2D paths is not valid. The sequence is composed of 30 scans of about 191,000 points.

Metrics: As in [23], the accuracy of the tested metrics is evaluated with the Euclidean distance Δ_t between the estimated transformation and the ground truth for translation and the geodesic distance Δ_r for rotation:

$$\Delta_t = \|^t\hat{\mathbf{t}}_s - ^t\mathbf{t}_s^*\| \qquad (29)$$

$$\Delta_r = arccos\left(\frac{trace(^t\mathbf{R}_s^{*-1}\,^t\hat{\mathbf{R}}_s) - 1}{2}\right) \qquad (30)$$

with $^t\hat{\mathbf{t}}_s$ and $^t\hat{\mathbf{R}}_s$ the estimated translation and rotation, $^t\mathbf{t}_s^*$ and $^t\mathbf{R}_s^*$ the ground truth translation and rotation respectively.

The thresholds to estimate a successful registration are 0.1m for translation and 2.5° for rotation as suggested in [7]. Note that rotation and translation errors are presented separately but a result is valid only if both rotation and translation errors are smaller than their respective threshold.

A. Impact of several minimization steps

This experiment aims to show the need for the point-to-plane registration step in NAP-ICP algorithm. As can be seen in Fig. 2 the estimation of the rigid transformation is performed in two successive steps. First a plane-to-plane minimization is performed (section III-D) followed by a point-to-plane registration (section III-E).

In Fig. 5, the impact of the steps can be observed with curves representing the cumulative probabilities errors on translation and rotation. The more top-left the curve the better the algorithm performs. The expected behavior is to attain 1 (meaning all scans of the sequence are registered) before the error threshold is reached. If so, it means that 100% of the scans of the sequence are successfully registered (according to the threshold previously). If 1 is not reached before the threshold, it means that the registration error is too large to be considered successful. On the plots, one can observe that for each sequence, the plane-to-plane registration gives a good initialization of the rigid transformation but is still far from ground truth. For instance, considering the *Apartment* sequence, only 41% scans are well registered regarding translation error (81% success rate in rotation). On *ETH* and *Stairs*, regarding rotation, even if the plane-to-plane registration gives results sufficient to reach the expected threshold, the addition of the point-to-plane proves to give a more accurate estimation.

For each tested sequence, the point-to-plane ICP step addition achieves a 100% success rate in rotation and translation.
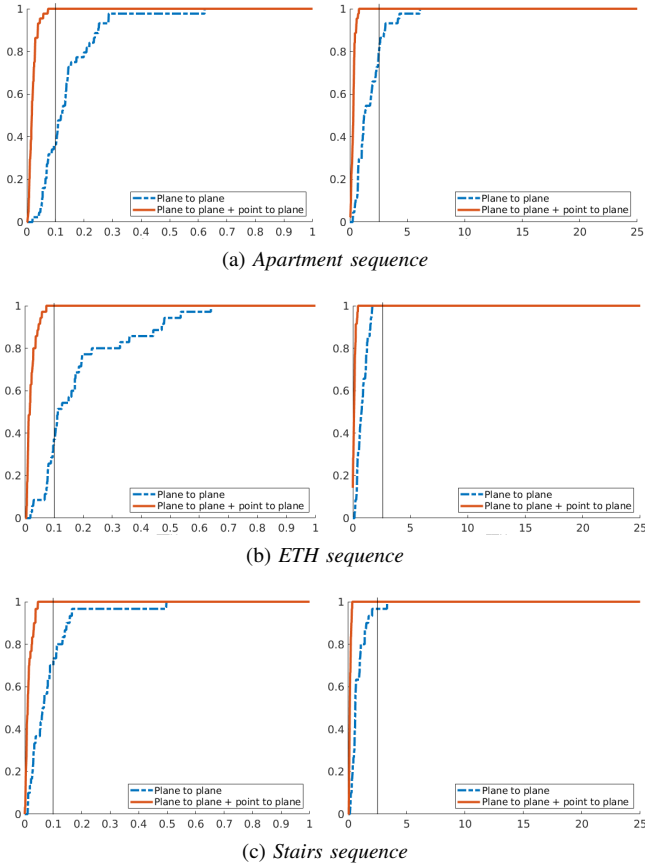
(a) *Apartment sequence*

(b) *ETH sequence*

(c) *Stairs sequence*

Fig. 5. Cumulative probabilities of translation and rotation errors for each step addition of the algorithm. *Left:* translation error (in meters) on the horizontal axis. The vertical bar represents a threshold (0.1m) for successful registration. *Right:* rotation error (in degrees) on the horizontal axis. The vertical bar represents a threshold (2.5°) for successful registration.

This proves the ability of the plane-to-plane minimization to give a result close enough to what is expected in order to obtain an accurate registration with a robust point-to-plane registration.

In Figure 6 all scans of *Apartment* sequence are registered in a mapping intention with the full algorithm. Each scan is registered with the previous one using the previously found transformation as initialization[1].

B. Comparison with state-of-the-art Algorithms

In the following experiment, NAP-ICP is compared to three state-of-the-art registration algorithms in terms of accuracy and computation time.

- G-ICP [8] has three major parameters. Maximum iterations is set to 10, Euclidean fitness epsilon is set to 10^{-6} and maximum correspondence distance is set to 0.8m.
- NDT, with the steps recommended in [7]. Transformation epsilon is set to 10^{-3}, step size 0.1, maximum iteration 5, first step resolution 1.0m, second step resolution 2.0m, third step resolution 1.0m and last step 0.5m.

[1]A short video presenting the mapping process can be found at: https://youtu.be/CL9gulE68rU
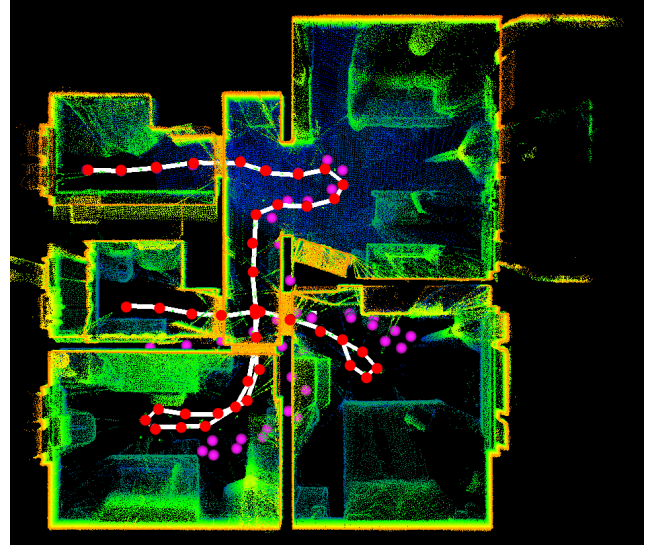

Fig. 6. 3D mapping of the *Apartment* sequence using the proposed method. The points of the map are colored regarding their height (blue being the lowest value, green in between and yellow the highest). In white the ground truth trajectory. In purple dots the trajectory computed with the plane-to-plane registration only. In red dots the trajectory computed with the full proposed NAP-ICP (combination of plane-to-plane and point-to-plane registration). Ceiling has been removed to ease visualization.

- Point-to-plane ICP [4], with the PCL implementation (denoted ICP-PCL), similarly to G-ICP has three major parameters. Maximum iterations is set to 100, Euclidean fitness epsilon is set to 10^{-6} and maximum correspondence distance is set to 0.8m.

Accuracy: To evaluate accuracy, equations (29) and (30) are used with all tested algorithms. The results of this experiment are summarized in Table I.

Globally, the proposed NAP-ICP algorithm gives more accurate results than G-ICP, NDT and ICP-PCL. On *Apartment*, *ETH* and *Stairs* sequences NAP-ICP achieves a 100% rate of successful registration. G-ICP, NDT and ICP-PCL give a 100% success rate on *ETH* sequence. However, their results on *Stairs* are not as good, even if still satisfying. On *Apartment* NAP-ICP outperforms the state-of-the-art algorithms. G-ICP achieves 75% of successful registrations, 77% for NDT and only 43% for ICP-PCL. This sequence includes large rotations (38% of the sequence is composed of motion with more than $\pm35°$ rotation on yaw axis) and G-ICP, NDT and ICP-PCL

TABLE I
PERCENTAGE OF SUCCESSFUL REGISTRATION (TRANSLATION AND ROTATION COMBINED) FOR THE EVALUATED ALGORITHMS ON EACH SEQUENCE

Sequence	NAP-ICP	G-ICP	NDT	ICP-PCL
Apartment	**100**	75	77	43
ETH	**100**	100	100	100
Stairs	**100**	97	97	90

sometimes struggle to find the right solution when our method succeeds. More detailed results with rotation and translation errors are presented in Fig. 7 for all sequences with curves representing the cumulative probabilities errors on translation and rotation for each tested algorithm. The most significant
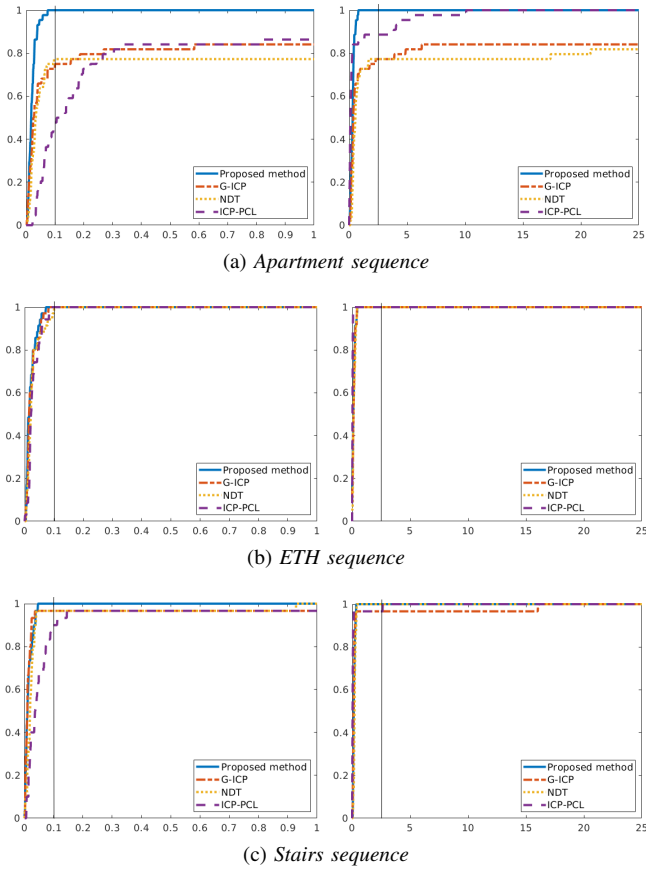
Sequence	NAP-ICP	G-ICP	NDT	ICP-PCL
Apartment	500	1790	**233**	339
ETH	1000	1800	**484**	808
Stairs	360	1300	**211**	375

V. CONCLUSION AND PERSPECTIVES

In this paper, NAP-ICP, an efficient plane-based registration algorithm for indoor 3D point clouds, is presented. This proposed method is based solely on LiDAR data.

A new metric based on planes characteristics is proposed to efficiently find the best plane correspondences. The robust plane-to-plane minimization followed by a point-to-plane minimization reaches 100% of successful registration on the tested sequences. Experiments show that NAP-ICP performs better than other state-of-the-art algorithms in well-structured environments (more specifically with large motion initialization between scans). They also showed that NAP-ICP algorithm is not only accurate but also fast. As it was not the main goal of this study, there is still room for optimization.

A more thorough study about weighting parameters for plane matching will be proposed in future works. This method also needs to be improved to handle outdoor urban environments.

Fig. 7. Cumulative probabilities of translation and rotation errors for each sequence on each evaluated algorithm. *Left:* translation error (in meters) on the horizontal axis. The vertical bar represents a threshold (0.1m) for successful registration. *Right:* rotation error (in degrees) on the horizontal axis. The vertical bar represents a threshold (2.5°) for successful registration.

feature of the proposed method in this experiment is its robustness to large motion scenarios (especially rotations) in comparison with other algorithms.

Computation time: No speed optimization are performed in NAP-ICP, however it is important to estimate the performances of the proposed method at this point. The experiments were held on a desktop computer with an Intel Xeon W-2133, 3.6GHz CPU and 32GB RAM. Processing time for the tested algorithms on each sequence is detailed in Table II. For each method, it includes all steps from point clouds preprocessing to transformation estimation.

On all sequences NDT is the fastest algorithm, followed by ICP-PCL. NAP-ICP method is slower than the aforementioned algorithms but compensate with its accuracy. G-ICP is the slowest method to handle this dataset.

REFERENCES

[1] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 239–256, Feb. 1992.

[2] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm," in *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, pp. 145–152, May 2001.

[3] F. Pomerleau, F. Colas, and R. Siegwart, "A Review of Point Cloud Registration Algorithms for Mobile Robotics," *Foundations and Trends in Robotics*, vol. 4, pp. 1–104, May 2015.

[4] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image and Vision Computing*, vol. 10, pp. 145–155, Apr. 1992.

[5] K.-l. Low, "Linear least-squares optimization for point-toplane ICP surface registration," tech. rep., Department of Computer Science, University of North Carolina at Chapel Hill, 2004.

[6] A. Fitzgibbon, "Robust Registration of 2D and 3D Point Sets," *Image and Vision Computing*, vol. 21, Jan. 2003.

[7] M. Magnusson, N. Vaskevicius, T. Stoyanov, K. Pathak, and A. Birk, "Beyond points: Evaluating recent 3D scan-matching algorithms," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3631–3637, May 2015.

[8] A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," *Proc. of Robotics : Science and Systems*, vol. 2, p. 4, 2009.

[9] Y. Taguchi, Y. Jian, S. Ramalingam, and C. Feng, "Point-plane SLAM for hand-held 3D sensors," in *2013 IEEE International Conference on Robotics and Automation*, pp. 5182–5189, May 2013.

[10] W. S. Grant, R. C. Voorhies, and L. Itti, "Efficient Velodyne SLAM with point and plane features," *Autonomous Robots*, vol. 43, pp. 1207–1224, June 2019.

[11] J. Poppinga, N. Vaskevicius, A. Birk, and K. Pathak, "Fast plane detection and polygonalization in noisy 3D range images," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3378–3383, Sept. 2008.

[12] K. Pathak, N. Vaskevicius, J. Poppinga, M. Pfingsthorn, S. Schwertfeger, and A. Birk, "Fast 3D mapping by matching planes extracted from range sensor point-clouds," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1150–1155, Oct. 2009.

[13] T. Rabbani, F. A. van den Heuvel, and G. Vosselman, "Segmentation of point clouds using smoothness constraints," in *ISPRS 2006 : Proceedings of the ISPRS commission V symposium Vol. 35, part 6 : image engineering and vision metrology, Dresden, Germany 25-27 September 2006*, pp. 248–253, 2006.

[14] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, pp. 381–395, June 1981.

[15] K. Pathak, A. Birk, N. Vakeviius, and J. Poppinga, "Fast Registration Based on Noisy Planes With Unknown Correspondences for 3-D Mapping," *IEEE Transactions on Robotics*, vol. 26, pp. 424–441, June 2010.

[16] S. Chen, L. Nan, R. Xia, J. Zhao, and P. Wonka, "PLADE: A Plane-Based Descriptor for Point Cloud Registration With Small Overlap," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, pp. 2530–2540, Apr. 2020.

[17] W. Zong, M. Li, Y. Zhou, L. Wang, F. Xiang, and G. Li, "A Fast and Accurate Planar-Feature-Based Global Scan Registration Method," *IEEE Sensors Journal*, vol. 19, pp. 12333–12345, Dec. 2019.

[18] F. Pomerleau, M. Liu, F. Colas, and R. Siegwart, "Challenging data sets for point cloud registration algorithms," *The International Journal of Robotics Research*, vol. 31, pp. 1705–1711, Dec. 2012.

[19] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *2011 IEEE International Conference on Robotics and Automation*, (Shanghai, China), pp. 1–4, May 2011.

[20] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-d point sets," *IEEE transactions on pattern analysis and machine intelligence*, vol. 9, pp. 698–700, May 1987.

[21] A. Lorusso, D. W. Eggert, and R. B. Fisher, "A Comparison of Four Algorithms forEstimating 3-D Rigid Transformations," *1995 BMVC*, vol. 1, pp. 237 –246, 1995.

[22] A. Comport, E. Marchand, M. Pressigout, and F. Chaumette, "Real-time markerless tracking for augmented reality: the virtual visual servoing framework," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 4, pp. 615–628, 2006.

[23] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing ICP variants on real-world data sets Open-source library and experimental protocol," *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, 2013.