



HAL
open science

Plane-based Accurate Registration of Real-world Point Clouds

Ketty Favre, Muriel Pressigout, Eric Marchand, Luce Morin

► **To cite this version:**

Ketty Favre, Muriel Pressigout, Eric Marchand, Luce Morin. Plane-based Accurate Registration of Real-world Point Clouds. SMC 2021 - IEEE International Conference on Systems, Man, and Cybernetics, Oct 2021, Melbourne / Virtual, Australia. pp.2018-2023, 10.1109/SMC52423.2021.9658727 . hal-03329646

HAL Id: hal-03329646

<https://inria.hal.science/hal-03329646>

Submitted on 31 Aug 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Plane-based Accurate Registration of Real-world Point Clouds

Ketty Favre[†], Muriel Pressigout^{*}, Eric Marchand[‡] and Luce Morin^{*}

Abstract—Traditional 3D point clouds registration algorithms, based on Iterative Closest Point (ICP), rely on point matching of large point clouds. In well-structured environments, such as buildings, planes can be segmented and used for registration, similarly to the classical point-based ICP approach. Using planes tremendously reduces the number of inputs.

In this article, an efficient plane-based registration algorithm is presented. The optimal transformation is estimated through a two-step approach, successively performing robust plane-to-plane minimization and non-linear robust point-to-plane registration. Experiments on the Autonomous Systems Lab (ASL) benchmark 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 robust in large motion scenarios and more accurate than other state-of-the-art algorithms. Moreover, a new challenging dataset, LOOP'IN, is provided. It is composed of two loops in real-world indoor scenes, with a large number of scans captured with a 3D LiDAR. Tests led on this dataset show that the algorithm is able to register long sequences, to close loops and to build an incremental map of the explored environment.

I. INTRODUCTION

The advances on LiDAR technology are no longer to be demonstrated. LiDAR sensors allow to obtain 3D data of the surrounding environment, represented as point clouds. The point clouds are used in many kinds of computer vision and robotics applications as localization and scene reconstruction, where registration of acquired 3D point clouds is still a widely studied problem with many challenges.

One of the most popular approaches to register 3D point clouds is the well-known Iterative Closest Point (ICP) algorithm [1]. It computes 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 within an iterative scheme until the residual error has reached the desired threshold.

ICP-like registration can be divided in two categories: local features and global features registration. The early approaches of the ICP algorithm were mostly using the local features. [1] uses the point-to-point distance and [2] the point-to-plane distance, proven to be more robust and to converge faster than the point-to-point one. A linear resolution of the point-to-plane distance minimization can be found in [3] using the small angle approximation. A non-linear Levenberg-Marquardt resolution using robust estimators is

proposed in [4]. Local registration methods may also extract keypoints from point clouds to establish correspondences. In [5], Normal Distribution Transform (NDT) takes into account local surface structures around each point. In Generalized-ICP (G-ICP) [6], the local neighborhood of points is used in order to assimilate this structure to small planar patches. The local normals of the source and target point clouds are taken into account like point-to-plane approaches. These methods provide satisfying performances but are usually time consuming when it comes to register large point clouds.

One way to overcome the point number issue is to use global features, such as planes. In such global registration approaches, the first step is to segment the point clouds into planar patches [7] [8] [9] and then use the planar features to establish correspondences. To match the planar patches, several approaches can be found. In [10], the data rate acquisition is considered high, which leads to low relative translation from one scan to the next. Thus, to determine plane correspondences, planes that are close to each other and almost parallel are matched. This approach is successful in this type of scenario, but struggles for recovering from large mistakes in the estimated trajectory. In [8], a plane/line descriptor is proposed to establish structure correspondences. It leads to efficient coarse registration. In [9], planar patches shape, area and normal are considered in order to find the best correspondences. In [11], a 3D mapping based on plane-segments is proposed.

The algorithm proposed in this article, Plane-based Accurate Registration ICP (PAR-ICP), stands in both global and local features categories. A two-level feature registration is performed, first using planes, then using points.

The main contributions of this article are:

- an algorithm performing accurate plane-based registration in challenging datasets and able to reconstruct the environment via an incremental map;
- a two-step minimization method with two levels of feature registration robust to large motion or inaccurate initialization;
- an efficient plane matching performed via classification using Random Forest (RF);
- an evaluation and a comparison of the proposed method with state-of-the-art algorithms on a challenging dataset;
- a new dataset, containing long sequences, and loops, for the point cloud community.

In the following sections, first, an overview of the framework of the proposed method is described. Then, the definitions and notations used in this article are detailed. After, plane matching using learned classification and distance minimization are presented in dedicated sections. Then,

[†]Univ Rennes, CNRS, IETR - UMR 6164, Rennes, France.
ketty.favre@univ-rennes1.fr

^{*}Univ Rennes, INSA Rennes, CNRS, IETR - UMR 6164, Rennes, France.

[‡]Univ Rennes, Inria, CNRS, IRISA, Rennes, France.

the provided dataset is succinctly described. Finally, the experiments and their results are presented.

II. PROPOSED ALGORITHM OVERVIEW

This section describes the framework of the proposed algorithm, named PAR-ICP. It is given in Fig. 1. Similarly to the classical ICP algorithm the method iteratively performs the matching step and minimization step. However, in the proposed method, plane matching and plane-to-plane distance minimization is first performed, followed by point matching and point-to-plane distance minimization. This last step aims to refine the estimation. An example of registration using the proposed method is given in Fig. 2.

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

- plane extraction and plane matching using classification are described in section IV;
- robust plane-to-plane registration is described in section V-A;
- the additional point-to-plane minimization leading to finer registration is detailed in section V-B.

III. ICP DISTANCE DEFINITIONS AND NOTATIONS

In the following sections, source and target points will be respectively denoted ${}^s\mathbf{P} = \{{}^s\mathbf{p}_i\}_{i=1}^N$ and ${}^t\mathbf{P} = \{{}^t\mathbf{p}_j\}_{j=1}^M$. In this registration problem, 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, defined as follows:

$${}^t\mathbf{T}_s = \begin{bmatrix} {}^t\mathbf{R}_s & {}^t\mathbf{t}_s \\ \mathbf{0}_{3 \times 1} & 1 \end{bmatrix} \quad (1)$$

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

For point-based registration, each point ${}^s\mathbf{p}_i$ in the source is matched to 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

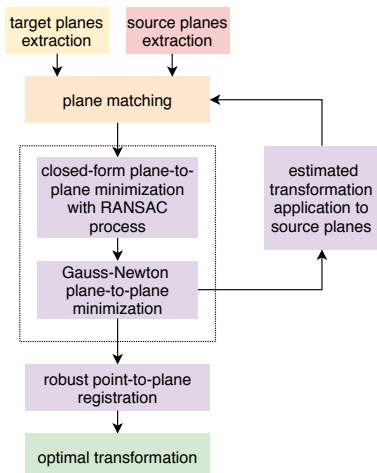


Fig. 1. PAR-ICP algorithm overview.

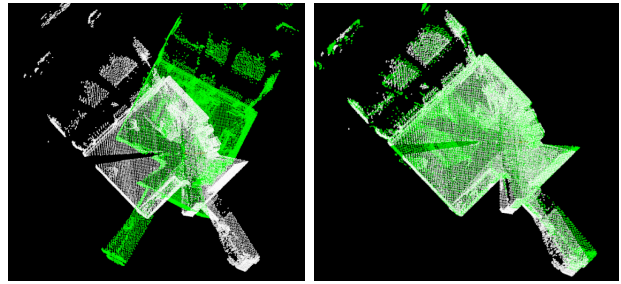


Fig. 2. Example of registration between two point clouds (scans 8 and 9 from *Apartment* sequence from ASL dataset [12]). The rotation between scans is large (more than 70° on yaw axis), yet the proposed method succeeds in registering the two point clouds accurately. In white the target point cloud - In green the source point cloud. *Left*: before registration - *Right*: after registration.

original ICP [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 [2], to minimize the point-to-plane distance (Fig. 3), defined by:

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

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

Regarding plane-to-plane registration, a plane $\Pi(\rho, \mathbf{n})$ is defined 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 ${}^s\Pi_i({}^s\rho_i, {}^s\mathbf{n}_i)$ the source plane and ${}^t\Pi_i({}^t\rho_i, {}^t\mathbf{n}_i)$ the target one, is given as follows [10] [13]:

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

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.

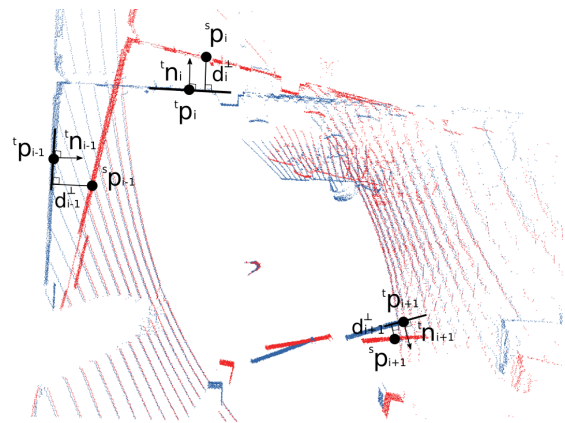


Fig. 3. Point-to-plane distance d_i^\perp as in [3]. In blue, the target point cloud and their related surface normals. In red the source point cloud to register.

IV. LEARNED PLANE MATCHING

To reduce the complexity of the pairing phase, thus, to reduce the number of potential correspondent planes, a Random Forest (RF) classifier is trained to find the best source/target plane pairs. Planar structures are extracted from the point cloud using a region growing segmentation algorithm based on [14], thanks to the Point Cloud Library (PCL) [15]. 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 that represents a plane. The point normals are estimated using a robust estimator based on iterative Principal Component Analysis (PCA) on the neighborhood of the concerned points [16]. For each extracted plane ${}^s\Pi_i$ in the source, a list of planes in the target that are potential correspondences is built.

A. Random Forest principle

RF classification [17] is a classical ensemble tree-based learning algorithm. It predicts the class of a target instance from values of several input variables, named features. A RF classifier gathers the predictions of many decision trees into a single model to take a decision. The training set is composed of n instances. Each instance is represented as a vector of m features f_1, \dots, f_m and a class label. First, k samples are built from randomly chosen subsets of the original training set using the bootstrapping principle. Then, k decision trees are built based on the k bootstrap samples. Once the model is trained, it is ready to predict classes of new instances. The k trees give a prediction. The class with the highest number of votes wins. By combining several predictions of de-correlated decision trees the model is less prone to overfitting than a single decision tree. RF classification was chosen for its low computational cost and its efficiency even with a small set of samples.

B. Training data and data augmentation

To train the classifier, a set of labeled data is needed. Pairs of planes are extracted from the indoor sequences of the ASL dataset [12] and manually labeled. Gaussian noise is added to the point clouds in order to use data that are not exactly the same as the original data. As this dataset provides the ground truth for each scan, it is possible to easily match planes from one scan to the next. The chosen classes are *Match* if the planes are potential correspondences and *Not-Match* otherwise.

In order to train the model with a larger set, data augmentation is performed on the previously labeled data, by applying rotations with angles randomly chosen in $[-70^\circ; +70^\circ]$.

C. Feature selection

The classifier is given the following features as input:

- The distance d_o between the projections of the origin on source plane and target plane:

$$d_o = \|\rho_i {}^s\mathbf{n}_i - \rho_j {}^t\mathbf{n}_j\|^2 \quad (4)$$

Planes are expected to be close to each other, so d_o is expected to be small.

- The distance d_c between the centroids of source and target planes:

$$d_c = \|\bar{\mathbf{p}}_i - \bar{\mathbf{p}}_j\|^2 \quad (5)$$

with ${}^s\bar{\mathbf{p}}$ and ${}^t\bar{\mathbf{p}}$ the centroids of ${}^s\Pi_i$ and ${}^t\Pi_j$ respectively. Once more, this feature aims to characterize the closeness of the planes, so d_c is expected to be small.

- The area ratio between the planes S_r :

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

with sS_i and tS_j the area of source and target planes respectively. Planes are expected to have similar shape, so S_r is expected to be close to 1.

- The dot product of the normals of the planes ϕ_n :

$$\phi_n = {}^s\mathbf{n}_i \cdot {}^t\mathbf{n}_j \quad (7)$$

Planes are expected to be close to parallel so ϕ_n is expected to be close to 1. This feature assumes small rotation between two scans.

D. Results on matching through classification

The classifier is evaluated through its accuracy, defined as the percentage of valid matches compared to ground truth. A 3-fold cross-validation process was used: two sequences were used for training while the third was kept for testing. The accuracy for each tested sequence is presented in Table I. As can be seen, the classifier produces some false matches, but their occurrence is less than 15% on the tested data. Such a percentage of outliers is manageable for a robust pose estimation method, as will be seen in next section.

V. REGISTRATION

A. Plane-to-plane registration

Once the set of plane correspondences is built, the plane-to-plane distance minimization to estimate the rigid transformation linking source to target planes, can be performed.

1) Closed-form plane-to-plane optimization method:

The derivation of this method is similar to the one presented in [13]. To take into account the presence of false matches in the list of corresponding planes, a RANSAC robust algorithm is used. RANSAC needs three non-parallel planes in the source and target respectively.

TABLE I

3-FOLD CROSS-VALIDATION RESULTS OF THE PAIRING PLANE PROCESS

k	Apartment	ETH	Stairs	Accuracy (%)
1	<i>testing</i>	training	training	87.9
2	training	<i>testing</i>	training	89.3
3	training	training	<i>testing</i>	91.6

2) Gauss-Newton plane-to-plane minimization method:

In order to refine the closed-form solution, the plane correspondences identified as inliers by the RANSAC algorithm are given as input of the non-linear iterative 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:

$$\hat{\mathbf{q}} = \underset{\mathbf{q}}{\operatorname{argmin}} \sum_{i=1}^N \|\mathbf{d}_i^\Pi\|^2 \quad (8)$$

where \mathbf{d}_i^Π is defined in by(3). Equation (8) can be solved using the Gauss-Newton algorithm. Solving it consists in minimizing the cost function $E(\mathbf{q}) = \|\mathbf{e}(\mathbf{q})\|$ where:

$$\mathbf{e}(\mathbf{q}) = \begin{pmatrix} \mathbf{n}(\mathbf{q}) - \mathbf{n} \\ \rho(\mathbf{q}) - \rho \end{pmatrix} \quad (9)$$

with $\mathbf{n}(\mathbf{q}) = (\dots, {}^t\mathbf{R}_s {}^s\mathbf{n}_i, \dots)$, $\mathbf{n} = (\dots, {}^t\mathbf{n}_i, \dots)$, $\rho(\mathbf{q}) = (\dots, [{}^t\mathbf{R}_s {}^s\mathbf{n}_i]^\top {}^t\mathbf{t}_s + {}^s\rho_i, \dots)$ 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.

This minimization problem can be solved by an iterative least-squares approach which gives $\delta\mathbf{q} = -\lambda\mathbf{J}(\mathbf{q})^+\mathbf{e}(\mathbf{q})$ where λ is a scalar in $]0, 1]$ and $\mathbf{J}(\mathbf{q})^+$ is the pseudo-inverse of the Jacobian $\mathbf{J}(\mathbf{q})$ that is given in [18]. The pose is then updated at each iteration using the exponential map.

B. Point-to-plane registration

If an even more accurate registration is required, a finer step can be added to find the best expected rigid transformation, based on the original data of the point clouds, rather than on the planes extracted from them. To do so, a point-to-plane registration is added at the end of the process.

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). Solving it consists in minimizing the cost function $E(\mathbf{q}) = \|\mathbf{e}(\mathbf{q})\|$ where $\mathbf{e}(\mathbf{q}) = (d_1^\perp, \dots, d_n^\perp)^T$.

The principle is the same as section V-A.2 with:

$$\hat{\mathbf{q}} = \underset{\mathbf{q}}{\operatorname{argmin}} \sum_{i=1}^N \|d_i^\perp\|^2 \quad (10)$$

where d_i^\perp is given by equation (2).

The minimization is handled as in the previous paragraph thanks to a Gauss Newton algorithm (Jacobian is given in [18]). Nevertheless, if outliers are present in the dataset, the minimization estimation becomes unstable. Considering a robust minimization process [19] is thus necessary. The Gauss-Newton algorithm allows for considering M-estimators within an iterative reweighted least square method. The error to minimize is defined by $e_\rho(\mathbf{q}) = \mathbf{D} \mathbf{e}(\mathbf{q})$ where \mathbf{D} is a diagonal matrix containing weights that reflect the confidence in the data (see e.g. [19] for details).

VI. EVALUATION DATASETS

To evaluate the robustness and the accuracy of the proposed algorithm, two datasets are used. The first one is ASL dataset [12] and the second one, LOOP'IN dataset, is provided with this article.

A. Autonomous Systems Lab dataset

As the focus of this article is on the usage of plane features, only the indoor sequences of the ASL dataset [12] 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.

- *Apartment* sequence is designed to evaluate algorithm robustness to outliers coming from dynamic elements (e.g. moved furniture).
- *ETH* sequence aims to evaluate robustness of registration to repetitive elements.
- *Stairs* sequence aims to evaluate robustness to rapid variations in scanned volumes.

B. LOOP'IN dataset

To evaluate PAR-ICP on long sequences and loop closure, a challenging dataset, LOOP'IN¹, was created.

The data acquisition were performed using a LiDAR VLP-16 Puck Hi-Res. The LiDAR is mounted on a moving platform pushed by an operator. As the captures are sometimes performed in narrow areas, the LiDAR is tilted towards the floor, to make sure to capture floor points.

The proposed dataset is composed of two indoor sequences, each scan consisting of 25K to 30K points.:

- *Balcony Loop*: The sequence is composed of a loop with a crossing of the trajectory in the end. The loop is made around a balcony surrounding an almost square room. The sequence is composed of 142 scans.
- *Coffee Hall Loop*: The sequence is composed of a loop starting and ending at the same location, in a large coffee hall. Furniture is present as well as some moving people. One side of the hall is a large bay window (making reflections appear on scans). The sequence is composed of 171 scans.

VII. EXPERIMENTAL RESULTS

The experiments are led in two steps. First, PAR-ICP is compared to state-of-the-art algorithms on the ASL benchmark dataset regarding robustness and accuracy. Then, the ability of the proposed method to register long sequences as accurately as possible is evaluated using the presented LOOP'IN dataset.

A. Validation on ASL dataset

To evaluate the accuracy and robustness of the proposed method, experiments on the ASL indoor sequences presented previously are performed. The dataset comes with a ground truth for the pose of each point cloud, allowing a quantitative evaluation of the accuracy of the estimated pose.

¹LOOP'IN dataset is available online at https://github.com/kfavre/LOOP-IN_dataset.

1) *Metrics*: As in [20], the accuracy of the algorithms is evaluated with the Euclidean distance Δ_t (respectively the geodesic distance Δ_r) between the estimated transformation and the ground truth for translation (respectively for rotation):

$$\Delta_t = \|\hat{\mathbf{t}}_s - \mathbf{t}_s^*\| \quad (11)$$

$$\Delta_r = \arccos\left(\frac{\text{trace}({}^t\mathbf{R}_s^{*-1}\hat{\mathbf{R}}_s) - 1}{2}\right) \quad (12)$$

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

The thresholds to estimate a successful registration are chosen as suggested in [5], i.e. translation threshold is 0.1m and rotation threshold is 2.5° . Note that rotation and translation errors are presented separately for the sake of clarity, but a result is valid only if both rotation and translation errors are smaller than their respective threshold.

The point clouds are registered frame-to-frame.

2) *Evaluation and comparison with state-of-the-art algorithms*: First, PAR-ICP is compared to two state-of-the-art registration algorithms:

- G-ICP [6] has three major parameters. Maximum iterations is set to 10, Euclidean fitness epsilon to 10^{-6} and maximum correspondence distance to 0.8m.
- NDT, with the steps recommended in [5]. 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.

Note that in this context, for PAR-ICP, the 3-cross fold validation is used for the RF classifier establishing correspondences, in order to avoid using data that were already seen in the validation. When the algorithm is tested on *Apartment* it has been trained only on *ETH* and *Stairs*, and so forth for the three sequences. Accuracy of PAR-ICP, G-ICP and NDT on translation and rotation are summarized in Figure 4 with curves representing the cumulative probabilities errors on translation and rotation. The expected behavior of the curves is to attain 1 (meaning all scans of the sequence are registered) before the error threshold is reached. As shown by these curves, PAR-ICP succeeds in registering 100% of the scans according to the chosen thresholds, for all tested sequences. It thus outperforms the other state-of-the-art methods, especially on the Apartment sequences, where PAR-ICP succeeds in registering all the scans, while G-ICP and NDT only achieves 75% and 77% of successful registrations, respectively. This can be explained by the presence of large rotations in the Apartment sequence, demonstrating PAR-ICP robustness to this type of motion.

In [9], several algorithms are evaluated following the same procedure: FGR [7], GH-ICP [21] and the algorithm proposed in [9] further called PGR (Planar Global Registration). The results of these experiments are presented in Table II.

On *Apartment*, FGR, PGR and PAR-ICP all completely succeed. The average rotation and translation error are respectively 0.45° and 0.022m for PAR-ICP, while they are 1.126° and 0.028m for FGR, and 0.303° and 0.009m for

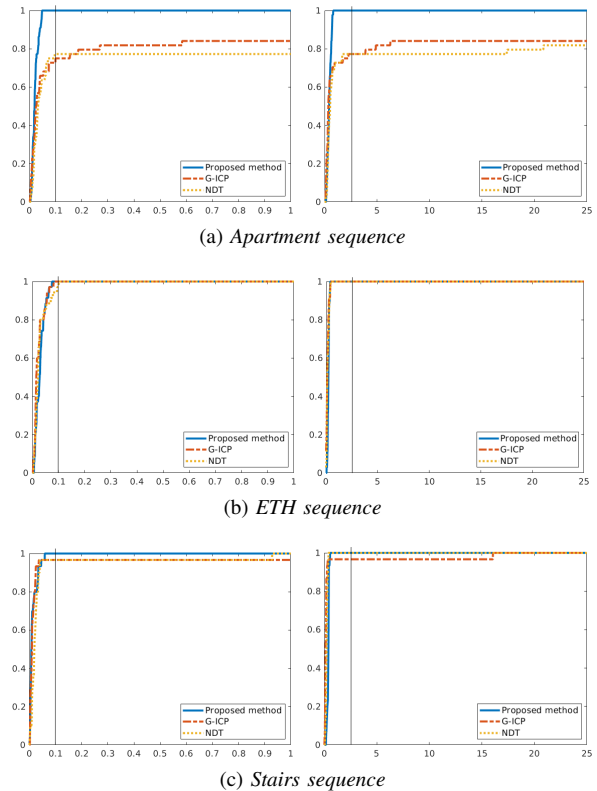


Fig. 4. Cumulative probabilities of translation and rotation errors for each sequence on PAR-ICP, G-ICP [6] and NDT [5]. *Left*: the translation error (in meters) on the horizontal axis. The vertical bar represents the threshold (0.1m) for successful registration. *Right*: the rotation error (in degrees) on the horizontal axis. The vertical bar represents the threshold (2.5°) for successful registration.

TABLE II

% OF SUCCESSFUL REGISTRATION (TRANSLATION AND ROTATION COMBINED) FOR THE ALGORITHMS EVALUATED IN [9]

Sequence	PAR-ICP	FGR [7]	GH-ICP [21]	PGR [9]
<i>Apartment</i>	100%	100%	52%	100%
<i>ETH</i>	100%	63%	93%	90%
<i>Stairs</i>	100%	3%	94%	97%

PGR. PGR ends up being the most accurate in this scenario. The reason why GH-ICP only achieves a 52% success rate is probably due to the presence of the large rotations in this sequence. On *ETH* and *Stairs* GH-ICP and PGR both give satisfying results but are still less accurate than PAR-ICP.

B. Registration in LOOP'IN dataset and incremental map creation using our approach

In the following experiment, the sensor trajectory of the LOOP'IN sequences are computed using PAR-ICP. In this context, all ASL sequences were used to train the RF classifier establishing correspondences. While each point cloud is being registered, a so-called incremental map is being built and updated. The registration of a new input source scan is not done frame-to-frame but by registering the scan to the incremental map. This allows to match source planes with

planes that were met earlier in the sequence, which gives important information in a long sequence, even more if a loop is performed. Planes in the source, that are already known and identified as matches in early steps, are updated in the map, and new unknown planes are added, expecting to be matched further in the reconstruction process. The trajectory on x and y axes for each sequence is presented in Fig. 5. In

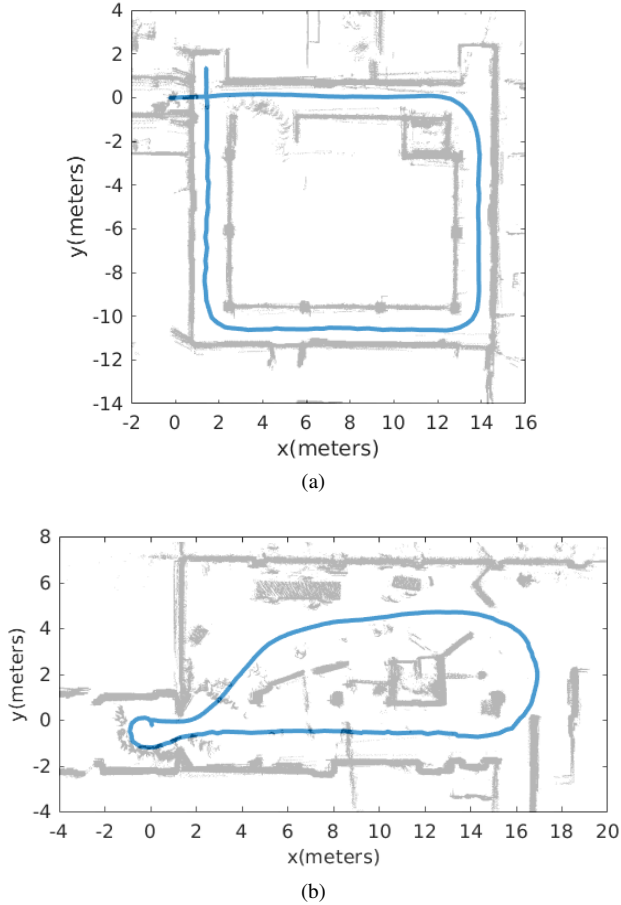


Fig. 5. Trajectory in xy-plane computed using PAR-ICP algorithm for each sequence. Axes are in meters. The reconstructed environment is superimposed on the trajectory by hand. (a) Balcony loop sequence (b) Coffee Hall sequence

both sequences, PAR-ICP is able to provide a satisfying path, and to close the loop between the starting and the ending poses, when using the incremental map for registration. Some inaccuracies in the trajectory can be observed from time to time, but they do not prevent the algorithm to close the loop and build a consistent global map of the environment.

VIII. CONCLUSION

In this paper, an accurate plane-based registration algorithm is presented. A plane matching approach using RF classifier trained on plane features is introduced. The proposed robust plane-to-plane minimization followed by a point-to-plane minimization reaches 100% of successful registration on the indoor sequences of the ASL dataset. Experiments show that PAR-ICP is more accurate than other state-of-the-art algorithms in well-structured environments. It also demonstrates to be robust to large motion.

A new challenging dataset, LOOP'IN, is also provided, for the community to evaluate and compare registration algorithms on real data. By performing loop closure in long sequences and creating coherent incremental maps, the proposed algorithm gives good results for localization and mapping applications.

A further step will be to handle a SLAM process with these two global and local levels of features.

REFERENCES

- [1] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on PAMI*, vol. 14, pp. 239–256, Feb. 1992.
- [2] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image and Vision Computing*, vol. 10, pp. 145–155, Apr. 1992.
- [3] K.-l. Low, "Linear least-squares optimization for point-to-plane ICP surface registration," tech. rep., Department of Computer Science, University of North Carolina at Chapel Hill, 2004.
- [4] A. Fitzgibbon, "Robust Registration of 2D and 3D Point Sets," *Image and Vision Computing*, vol. 21, Jan. 2003.
- [5] M. Magnusson, N. Vaskevicius, T. Stoyanov, K. Pathak, and A. Birk, "Beyond points: Evaluating recent 3D scan-matching algorithms," in *2015 IEEE ICRA*, pp. 3631–3637, May 2015.
- [6] A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," *Proc. of Robotics : Science and Systems*, vol. 2, p. 4, 2009.
- [7] Q.-Y. Zhou, J. Park, and V. Koltun, "Fast global registration," in *Computer Vision – ECCV 2016*, vol. 9906, 10 2016.
- [8] 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.
- [9] 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.
- [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] 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 *IEEE/RSJ IROS*, pp. 1150–1155, Oct. 2009.
- [12] 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.
- [13] Y. Taguchi, Y. Jian, S. Ramalingam, and C. Feng, "Point-plane SLAM for hand-held 3D sensors," in *IEEE ICRA '13*, pp. 5182–5189, 2013.
- [14] T. Rabbani, F. A. van den Heuvel, and G. Vosselman, "Segmentation of point clouds using smoothness constraints," in *ISPRS 2006 Vol. 35, part 6 : image engineering and vision metrology, Dresden, Germany 25-27 September 2006*, pp. 248–253, 2006.
- [15] 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.
- [16] J. Sanchez, F. Denis, D. Coeurjolly, F. Dupont, L. Trassoudaine, and P. Checchin, "Robust normal vector estimation in 3D point clouds through iterative principal component analysis," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 163, pp. 18–35, 2020.
- [17] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, pp. 5–32, Oct. 2001.
- [18] K. Favre, M. Pressigout, L. Morin, and E. Marchand, "A plane-based approach for indoor point clouds registration," in *IAPR Int. Conf on Pattern Recognition, ICPR'20*, (Milano (virtual), Italy), January 2021.
- [19] E. Malis and E. Marchand, "Experiments with robust estimation techniques in real-time robot vision," in *IEEE/RSJ IROS'06*, (Beijing, China), pp. 223–228, October 2006.
- [20] 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, pp. 133–148, 2013.
- [21] Z. Dong, B. Yang, Y. Liu, F. Liang, B. Li, and Y. Zang, "A novel binary shape context for 3d local surface description," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 130, pp. 431–452, 2017.