A Plane-based Approach for Indoor Point Clouds Registration

Ketty FAVRE - Muriel PRESSIGOUT - Eric MARCHAND - Luce MORIN

VAADER IETR research team

Motivations

In robotics, registration of 3D point sets is a key issue in localization applications. The trend for autonomous vehicles makes it a widely search field.

In this application the 3D point clouds are captured with a LiDAR and the registration is based solely on this information.

Iterative Closest Point (ICP) [1] is one of the mostly used algorithms for 3D point clouds registration, but, the point matching step is known to be time consuming due to the potential large number of points.

Objective

The aim of the presented algorithm is to find the transformation ${}^t\mathbf{T}_s$ that best fits a source point cloud to a target point cloud. The transformation is defined as follows:

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

with ${}^t\mathbf{R}_s$ and ${}^t\mathbf{t}_s$ respectively a 3x3 rotation matrix and a 3x1 translation vector.

Man-made environment, such as buildings, are usually composed of strong planar structures. Planes are less numerous than points and give a good representation of the captured scene. Thus, similarly to the classical ICP algorithm, the proposed algorithm iteratively performs a matching step and a minimization step, but instead of points, planes are registered.

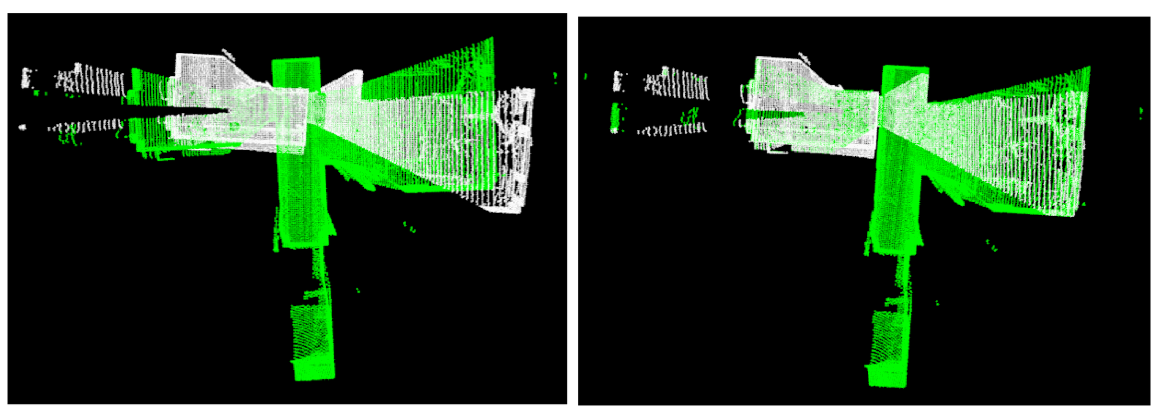


Fig.1 Example of a registration between two point clouds (scans from ASL dataset [2]). In white the target point cloud - In green the source point cloud. Left: before registration - Right: after registration.

Distances Definitions

To find ${}^t\mathbf{T}_s$, the plane-to-plane distance is minimized. To ensure an accurate registration, an additional point-to-plane registration is added at the end of the process (Fig.4). Both minimization are performed using a nonlinear Gauss-Newton optimization.

Plane-to-plane distance:

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

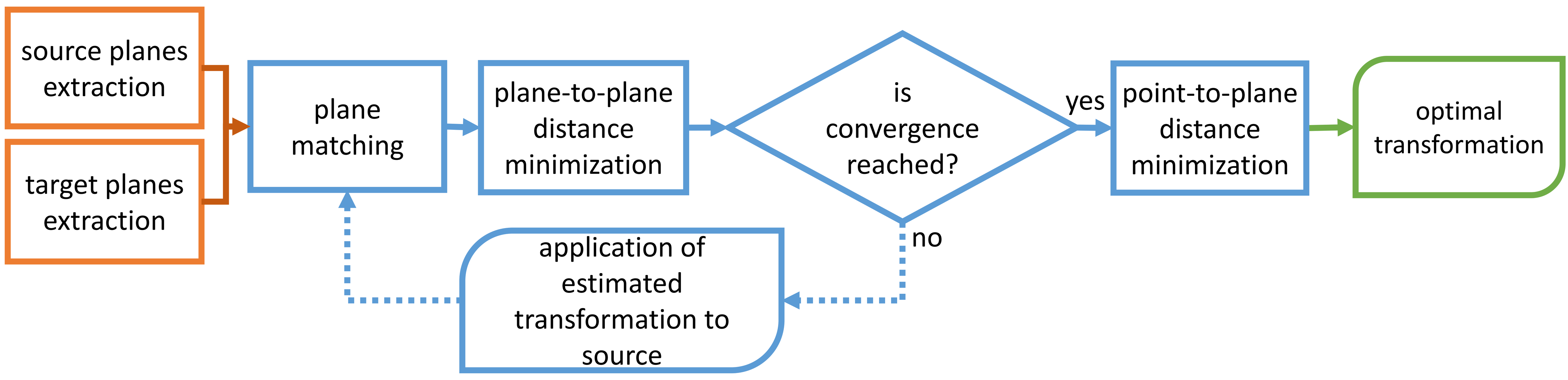
where ${}^s\mathbf{n}_i$ and ${}^t\mathbf{n}_i$ are the normal to the planes ${}^s\Pi_i$ and ${}^t\Pi_i$, respectively, and ${}^s\rho_i$ and ${}^t\rho_i$ their respective distance to the origin of the sensor in the target frame.

Point-to-plane distance:

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

with ${}^s\mathbf{p}_i$ and ${}^t\mathbf{p}_i$, respectively the source and target point, and ${}^t\mathbf{n}_i$ the surface normal computed from ${}^t\mathbf{p}_i$ neighborhood.

Algorithm Framework



Plane Extraction

Planes are extracted using a region growing segmentation based on [3]. 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.

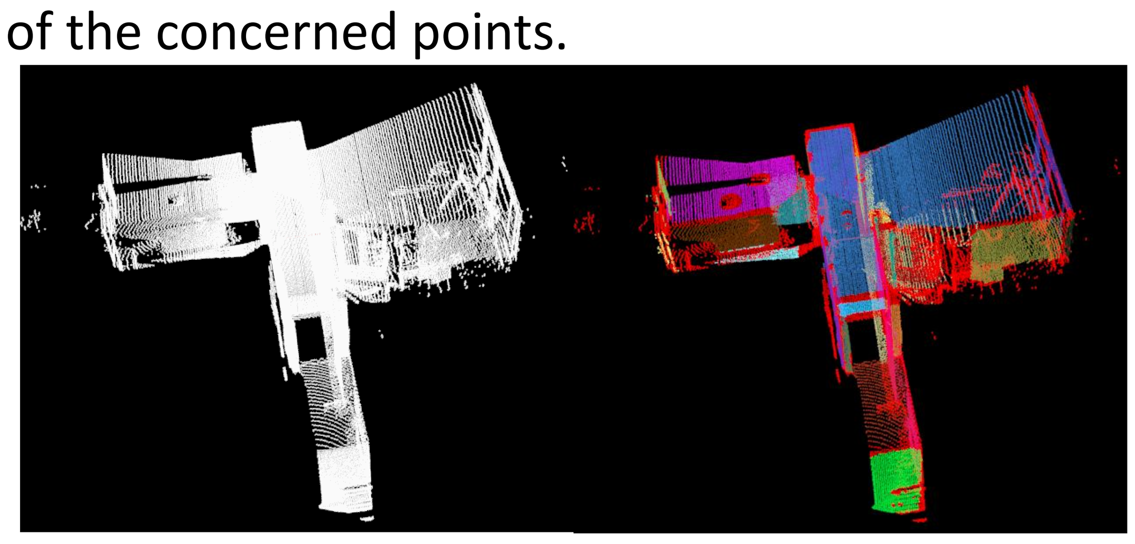


Fig.2 Plane segmentation example. Left: input point cloud - Right: plane extraction result. Each extracted plane is in a different color. Red points are outliers.

Plane Matching

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. 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 of the planes:

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

- the distance between the centroids of the planes:

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

with ${}^s\bar{\mathbf{p}}_i$ and ${}^t\bar{\mathbf{p}}_j$ the centroids of source end target planes.

- the area ratio between the planes:

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

- the dot product of the normals of the planes:

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

Each feature is normalized between [0,1] (denoted $\hat{}$ further) and weighted, leading to a score defined as follows:

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

with: $$ \alpha + \beta + \gamma + \delta = 1 $$

A target plane is considered as a valid match if it respects the following condition: $score < t_{score}$

Results

Comparison with state-of-the-art algorithms:

Using the indoor sequences of the Autonomous System Labs (ASL) [2] dataset, the distance of the estimated pose to the ground truth is evaluated with the Euclidean distance Δ_t and the geodesic distance Δ_r (Fig.3):

$$ \Delta_t = \| {}^t\hat{\mathbf{t}}_s - {}^t\mathbf{t}_s^* \| $$

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

With ${}^t\mathbf{R}_s^*$ and ${}^t\mathbf{t}_s^*$ respectively the ground truth rotation matrix and translation vector. The thresholds to estimate a successful registration are 0.1m for translation and 2.5° for rotation, as suggested in [4].

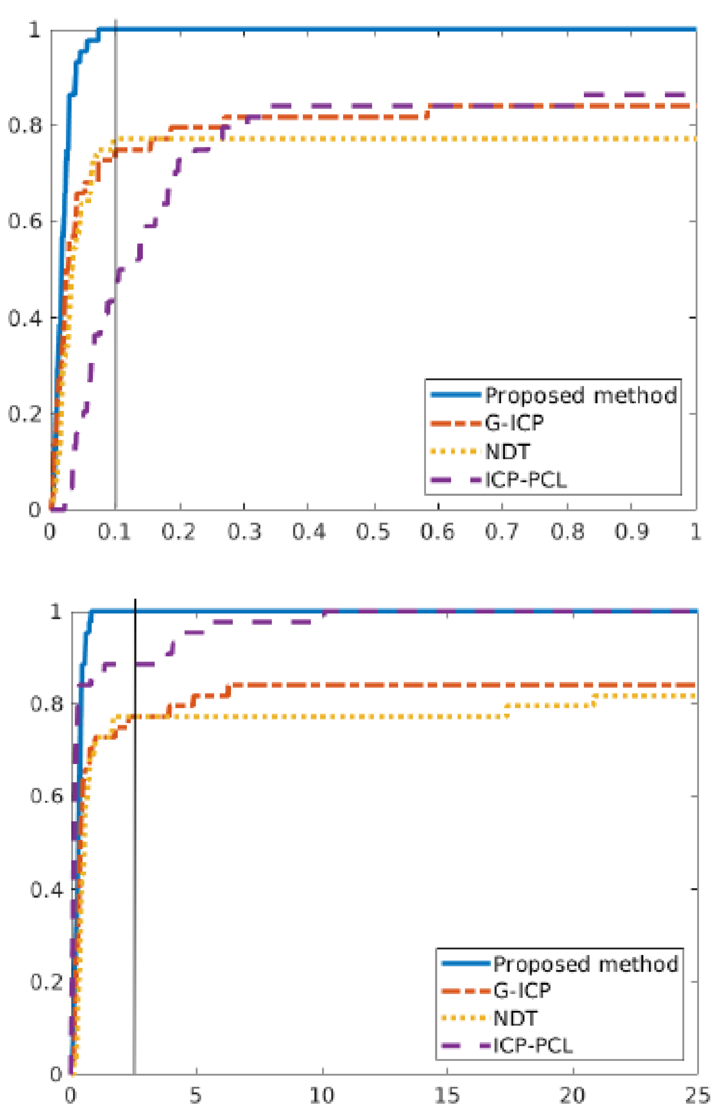


Fig.3 Cumulative probabilities of : (top) translation error (in meters) and (bottom) rotation error (in degrees) for Apartment sequence on each evaluated algorithm. The vertical bars represent the thresholds (respectively 0.1m for translation and 2.5° for rotation) for successful registration.

Tab.1 Percentage of successful registration (translation and rotation combined) for the evaluated algorithms on each considered sequence [2].

Sequence	Proposed method	G-ICP[5]	NDT[4]	ICP-PCL
Apartment	**100**	75	77	43
ETH	**100**	**100**	**100**	**100**
Stairs	**100**	97	97	90

Impact of the two-step minimization:

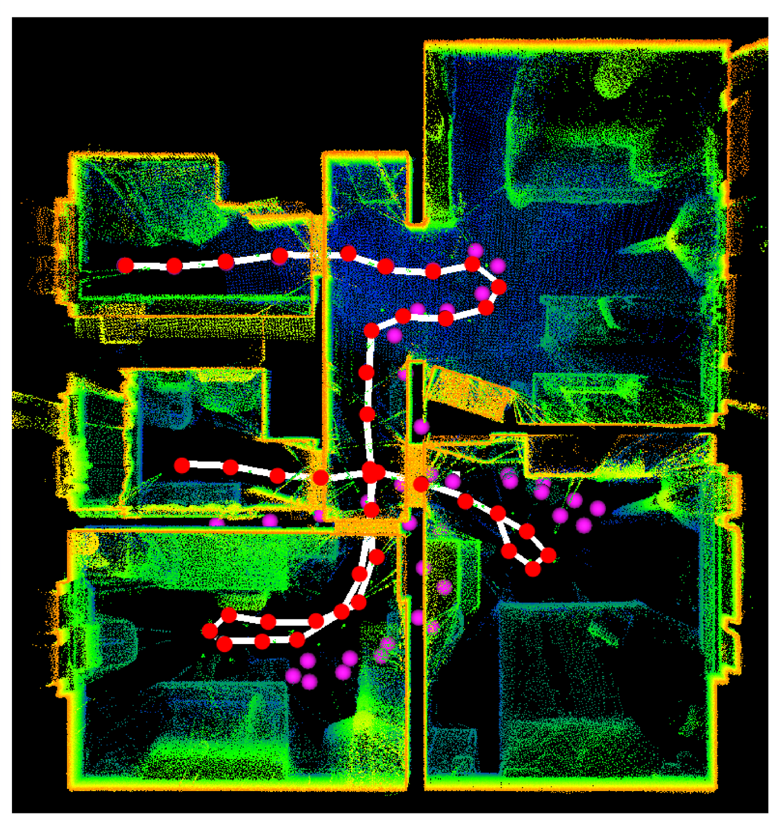


Fig.4 3D mapping of the Apartment sequence using the proposed method. White: the ground truth trajectory. Purple dots: plane-to-plane only registration trajectory. Red dots: combination of plane-to-plane and point-to-plane registration trajectory.

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] 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.

[3] 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 Vsymposium Vol. 35, part 6 : image engineering and vision metrology, Dresden, Germany 25-27 September 2006, pp. 248–253, 2006.

[4] 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.

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