American Academic Scientific Research Journal for Engineering, Technology, and Sciences

ISSN (Print) 2313-4410, ISSN (Online) 2313-4402

http://asrjetsjournal.org/

A New Variant of the ICP Algorithm for Pairwise 3D Point Cloud Registration

Elizeu Martins Oliveira Junior[a]*, Daniel Rodrigues Santos[b], Giovana Angélica Ross Miola[c]

[a]Geodesic Sciences Pos-graduate Program, Paraná Federal University, 100 Francisco H. dos Santos Avenue, Curitiba-PR, 81539-000, Brazil

[b]Cartographic Engineering Section, Military Institute of Engineering, 80 Gen. Tiburcio Square, Rio de Janeiro-RJ, 22290-270, Brazil

[c]Presidente Prudente Technology Faculty, Informatics Department, 17 Terezinha Street, Presidente Prudente – SP, 19049-230, Brazil

[a]Email: elizeuoliveira@ufpr.br

[b]Email: daniel.rodrigues@ime.eb.br

[c]Email: giovana_ros@hotmail.com

Abstract

Pairwise 3D point cloud registration derived from Terrestrial Laser Scanner (TLS) in static mode is an essential task to produce locally consistent 3D point clouds. In this work, the contributions are twofold. First, a non-iterative scheme by merging the SIFT (Scale Invariant Feature Transform) 3D algorithm and the PFH (Point Feature Histograms) algorithm to find initial approximation of the transformation parameters is proposed. Then, a correspondence model based on a new variant of the ICP (Iterative Closest Point) algorithm to refine the transformation parameters is also proposed. To evaluate the local consistency of the pairwise 3D point cloud registration is used a point-to-distance approach. Experiments were performed using seven pairs of 3D point clouds into an urban area. The results obtained showed that the method achieves point-to-plane RMSE (Root of the Mean Square Error) mean values in the order of 2 centimeters.

Keywords: Pairwise 3D Point Clouds Registration; Terrestrial Laser Scanner; Planar Surface; ICP.

----

* Corresponding author.

1. Introduction

In static mode, TLS (Terrestrial LASER Scanner) is a tool capable of providing 3D point clouds with high level of detail, quickly, accurately and safely. Given the technological advances in this field, TLS has been increasingly used in topographic surveys of the physical surface of the Earth for several of applications, such as mapping, surveillance and emergency managements, navigation, positioning, robotics, forensics, Earth science, virtual tours, crisis management, modeling, infrastructure inspections, urban design, archaeology, Civil Engineering and others.

Due to the scanning characteristics of the TLS, the complete overlay of an object present on the physical surface should be done with different views. For instance, consider a TLS device (S) that moves in an internal environment, such as the hallway of a building. At each position station ($S_1, \ldots, S_n$) the sensor collects thousands of 3D points from a small part of that environment. During the data acquisition, an accumulation of errors generated by sensor uncertainty is introduced and each 3D point cloud is obtained in an independent Local Referential System (LRS). The result is a set of 3D point clouds with angular misalignment and linear displacement between each other. Consequently, the registration of each pair of 3D point clouds (pairwise 3D point cloud registration) and the materialization of a unique reference system for the data set is a fundamental task to create a complete and accurate 3D model of the mapped surface.

In the specific literature, the pairwise 3D point cloud registration feature-based is divided into two steps: (1) the primitive detection and automatic establishment of the feature correspondences; (2) the estimation of the transformation parameters (usually a rotational matrix *R* and a tridimensional translation vector *t*).In [1] the first stage is classified into two categories: a) correspondence models based on point-to-plane approaches; and b) surface-based matching models.

According to [21], plane-based approaches provide better accuracy in estimates of transformation parameters. Moreover, conforming to [14], point-to-plane or plane-to-plane approaches are less susceptible to noise, besides being easily found in anthropic environments (man-made) and are robust to environments with homogeneous surfaces, such as facades. An iterative approach based on point-to-plane correspondence models are proposed in [23]. First, the algorithm selects defined points from the Newton-Raphson technique and calculates the normal vectors related to each of them. The determination of the correspondences is performed by an iterative process and the transformation parameters are also estimated in two steps. In the first one, the components of the rotational matrix are determined with quaternions properties. In the second one, the translation values are determined by LSM (Method of Least Squares) where the rotation parameters are fixed in the solution.

A plane-based correspondence from the surface curvature values analysis is performed in [5]. The transformation parameters are estimated by LSM and refined with Levenberg-Marquart method. In [16], the RANSAC algorithm (FISHER and BOLLES, 1981) is used for the plane extraction followed by a directions and angles (formed between their normal vectors analysis) strategy for automatic establishment of plane-to-plane correspondences. The estimation of transformation parameters is also performed in two steps. However, planes are common only in man-made environments, making this approach unadvised in natural environments or in environments with a high degree of symmetry between the objects present in the scene.

Besides the unpredictably of the objects available in the mapped environment, the widest used algorithm in literature for surface-based approaches still is the ICP (Iterative Closest Point)[17].Basically, the ICP algorithm is done in two stages: First, the algorithm establishes pseudo-matches between point cloud pairs. Then, the sum of the square of the distances between these pseudo-matches is iteratively minimized. And commonly, the estimated values of rotation (R) are used to calculate the translation parameters (t). This is done repeatedly until the algorithm reaches a convergence criterion. However, the ICP algorithm can converge to a local minimum solution when the point cloud pairs present a low rate of overlap [22]. Furthermore, the ICP depends on approximate initial values and has a high computational cost. Thus, in literature it is usually necessary to find initial approximations of the transformation parameters and, consequently, refine the transformation parameters with the ICP. This pairwise registration approach is known as coarse-to-fine [9].

The high computational cost problem of the ICP algorithm is overcome by [3] employing the k-d Tree technique in the correspondence step. To solve the same dependence on initial values of the ICP, [13], define initial alignments of the sensor using primitives extracted from the 3D point clouds pairs. The ICP-RGBD algorithm is developed in [18] for pairwise 3D point cloud registration derived from RGB-D data. First, the points in the RGB image are detected and their matches are established using the Scale Invariant Feature Transform (SIFT) 2D algorithm, proposed by [8]. Then, the image points are associated with their respective 3D points derived from the depth image of the RGB-D sensor. The transformation parameters are also refined with the ICP algorithm.

A local alignment method for pairwise 3D point cloud registration called 4-Points Congruent Sets (4PCS) is introduced in [6]. The alignment process occurs from the determination of the set with the best matches. The 4PCS method is adapted by [19] with the novel K-4PCS aiming to reduce the algorithm flow time processing (inferred as computational cost). The authors propose the extraction of key points from the Dog (Difference of Gaussians) operator, whose resamplings are made from structures called VOXELS (VOlume X ELementS). The transformation parameters are obtained by approximation and later refined using the ICP, i.e. in a coarse-to-fine mode.

The pairwise 3D point cloud registration methods also can be characterized by the rotation parameters estimation approaches. In [15] the rotation matrix is estimated by SVD (Singular Value Decomposition). Once in[4], unit quaternions are used for the rotational matrix components representation. According to [10], both SVD and quaternions uses approximation models to make the rotation estimation.

The main challenge in pairwise 3D point cloud registration by TLS, in static mode, is related to the low overlap percentage between pairs of 3D point clouds, since the operator seeks to reduce the in loco operational cost. Typically, TLS in static mode is not supported by additional sensors, such as GNSS/INS systems, so the estimation of parameters is never simplified, once does not have a very initial guess for the transformation parameters. Besides the amount of 3D point cloud pairs necessary to ensure the complete covering of the region mapped. With the motivation of producing locally consistent 3D point clouds for represent Earth’s physical surface, two contributions will be discussed in this work:

- A non-iterative strategy of initial approximation of 3D point cloud pairs; and
- A novel correspondence method based on an ICP algorithm new variant.

2. Proposed method

This work aims to deal with the pairwise registration problem of 3D point cloud obtained by a static TLS to build locally consistent 3D models. The proposed method is divided into four main tasks, as shown in Figure 1.

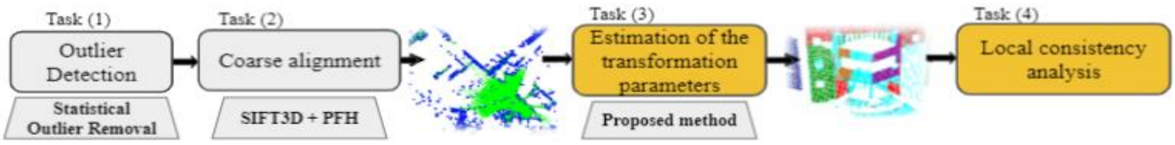
Task (1)
Outlier Detection
Statistical Outlier Removal
Task (2)
Coarse alignment
SIFT3D + PFH
Task (3)
Estimation of the transformation parameters
Proposed method
Task (4)
Local consistency analysis

Figure 1: Architecture of the proposed method

The first task is to detect and remove outliers present in the 3D point clouds. The second task is to reduce the data to a sparse set of points and calculates the initial approximate values for the 3D cloud pair. In task three, the 3D point cloud pairwise transformation parameters are estimated using a point-to-plane correspondence model proposed in this work. Finally, the analysis of local consistency (task four) of each pair of 3D point clouds is performed by a point-to-plane distance criterion. The expected result is locally consistent sets of 3D point clouds from the mapped environment. The steps of the proposed method are discussed below.

2.1. Detection and Removal of Outliers

The first task of the proposed method is automatically to detect and remove outliers present in the 3D point cloud pair. This is done using the Statistical Outlier Removal (SOR) algorithm proposed by [20]. Outliers are defined as observations in a data set that are inconsistent with the rest of this data set, usually generated by specular surfaces such as glass and metals, or also arising from the transition between two surfaces (edges and occlusions).

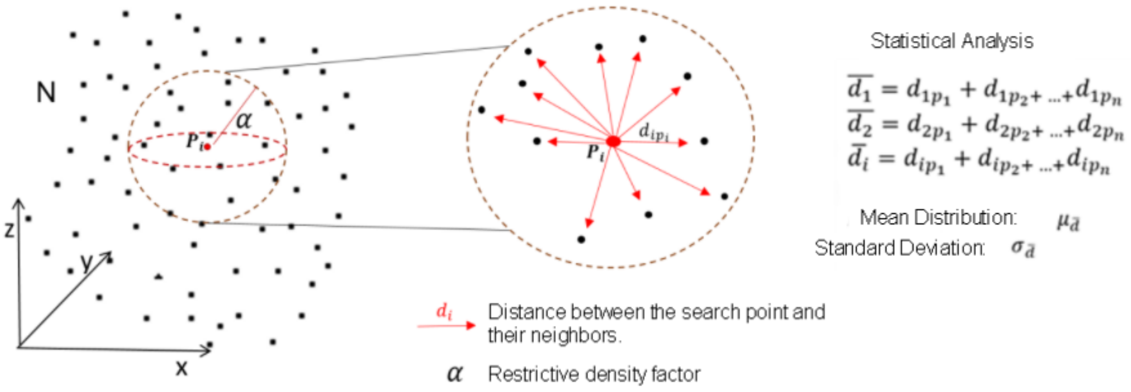
N
Z
Y
X
P_i
α
P_i
d_{ip_i}
Statistical Analysis
$\overline{d_1} = d_{1p_1} + d_{1p_2} + ... + d_{1p_n}$
$\overline{d_2} = d_{2p_1} + d_{2p_2} + ... + d_{2p_n}$
$\overline{d_i} = d_{ip_1} + d_{ip_2} + ... + d_{ip_n}$
Mean Distribution: $\mu_{\bar{d}}$
Standard Deviation: $\sigma_{\bar{d}}$
d_i Distance between the search point and their neighbors.
α Restrictive density factor

Figure 2: Process to find the nearest neighbor using Statistical Analysis.

Due to data acquisition characteristics, the transformation parameters estimation step is extremely sensitive to outliers. Therefore, these outlier removal tasks dramatically reduce computational cost. Hence, the most intuitive way to detect and remove outliers is by using neighborhood analysis of a random point present in the 3D point cloud, as proposed by [20], namely: a) For each point $p_i \in N$, firstly, the average distance $\underline{d}$ from its k nearest neighbors is calculated; b) Then, the average distribution of the points $\mu_{\underline{d}}$ and their standard deviation $\sigma_{\underline{d}}$ are estimated, with the objective of keeping in the point cloud N the points whose distance $\underline{d}$ to the nearest neighboring point is similar to the other points; c) Thus, the remaining point cloud N^* is determined as follows:

$$N^* = \{p_i^* \in N | (\mu_k - \alpha \cdot \sigma_k) \leq \underline{d}_k \leq (\mu_k + \alpha \cdot \sigma_k)\} \qquad (1)$$

where α is the restrictive density factor. Figure 2 shows the steps described above.

2.2. Coarse Pairwise 3D Point Cloud Registration

Usually, the TLS sensor operator seeks to optimize the work by installing TLS in positions that provide low overlap between point cloud pairs (30% to 40%), making it difficult to solve the ICP algorithm during the estimation of transformation parameters. In this work, a combination of the 3D SIFT algorithm and the PFH algorithm is proposed to find initial transformation parameters between the 3D point cloud pairs. The 3D SIFT algorithm is used to extract extreme points in the 3D point cloud pairs and the PFH algorithm builds local descriptors invariant to scale, rotation and change of view, in the form of representative neighborly relations histograms between the extreme points and their respective normal vectors for the automatic matching.

In 3D SIFT, the space-scale of a 3D point cloud is defined as a 4D function $L(x, y, z, \sigma)$ obtained by Gaussian Kernel $G(x, y, z, \sigma)$convolution with a point cloud $N(x, y, z)$, as follows [12]:

$$N(x, y, z, \sigma) = N(x, y, z) \otimes G(x, y, z, k \cdot \sigma) \qquad (2)$$

where $\otimes$ is the convolution operator, σ the scale change in G (parameter defining the smoothing factor) and $G(x, y, z, k \cdot \sigma) = \frac{1}{(\sqrt{2\pi}k \cdot \sigma)^3} e^{\frac{-(x^2+y^2+z^2)}{2(k \cdot \sigma)^3}}$. Figure 3 shows an example of space-scale in 3D.

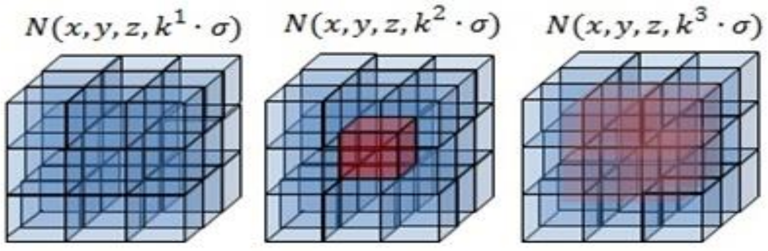
$N(x, y, z, k^1 \cdot \sigma)$
$N(x, y, z, k^2 \cdot \sigma)$
$N(x, y, z, k^3 \cdot \sigma)$

Figure 3: Example of space-scale in 3D. The blue cube represents a voxel.

In the sequence, Gaussian Difference calculations (Difference of Gaussians - Dog) are performed for each octave in the space-scale. This function is separated by a constant scale (k) and the keypoints in the 4D space-scale are detected as local extremes (maximum or minimum) of the N differences in nearby scales defined by $i \in [0, s+2]$, as follows [12]:

$$DoG(x,y,z,k^i \cdot \sigma) = N(x,y,z,k^{i+1} \cdot \sigma) - N(x,y,z,k^i \cdot \sigma) \quad (3)$$

The local extreme points can be detected for each $DoG(x,y,z,k^i \cdot \sigma)$ obtained. This procedure is based on comparing all voxels of the current $DoG(x,y,z,k^i \cdot \sigma)$ with their neighbors voxels according to the correspondent neighbors voxels in $DoG(x,y,z,k^{i+1} \cdot \sigma)$ and $DoG(x,y,z,k^{i-1} \cdot \sigma)$resulting 80 neighbors voxels in total (27 $(i+1)$ + 26 (i) + 27 $(i-1)$ = 80).

The extreme points must be located and if they are unstable, infers being discarded. The exact location of the extreme points is determined by adjusting a 3D quadratic function. The DoG function has a robust response along the edges, making the points unstable. This implies ill-defined extremes that exhibit large principal curvature along the edges, but with small curvature in their perpendicular direction. The principal curvatures are determined, basically, through the Hessian 3x3 (H) matrix, as follows [7]:

$$H(\hat{x},\sigma) = \left[S_{xx}(\hat{x},\sigma)S_{xy}(\hat{x},\sigma)S_{xz}(\hat{x},\sigma)S_{yx}(\hat{x},\sigma)S_{yy}(\hat{x},\sigma)S_{yz}(\hat{x},\sigma)S_{zx}(\hat{x},\sigma)S_{zy}(\hat{x},\sigma)S_{zz}(\hat{x},\sigma)\right] \quad (4)$$

where:

$S_{xx}(\hat{x},\sigma) = DoG(x,y,z,k^i \cdot \sigma) \otimes \frac{\partial^2}{\partial x^2} G(x,y,z,\sigma)$. Calculating $H(\hat{x},\sigma)$ on multiple scales and searching for local maxima, a set of extreme points X can be obtained, as follows [7]:

$$X = arg_{\hat{x},\sigma}|det\big(H(\hat{x},\sigma)\big)| \quad (5)$$

With the extreme points detected by the 3D SIFT algorithm, the normal vectors$\vec{n}_i$ of all extreme points $X_i \in N^*$must be estimated. In this case, a plane is represented by a point X (extreme point) in R^3 and a normal vector $\vec{n}$, and, the distance from a point $p_i \in N^*$ to the plane is defined as $d_i = (p_i - X)\vec{n}$. Since there is a set of neighbors points (p_i) circumscribed in a circle (sv_i) of radius r_1, the solution for $\vec{n}$ is obtained by analyzing the eigenvalues and eigenvectors of the covariance matrix $C \in R^{3x3}$ of sv_i, as follows:

$$C = \frac{1}{m}\sum_{i=1}^{m} W_i(p_i - X)(p_i - X)^T, C\vec{V}_j = \quad {}_j\vec{V}_j, j \in \{0,1,2\} \quad (6)$$

where C is a semi-definite positive symmetric matrix and its eigenvalues are real numbers $\lambda_j \in \mathbb{R}$. The eigenvectors $\vec{V}_j$ correspond to the principal components. If $0 \leq \lambda_0 \leq \lambda_1 \leq \lambda_2$, the eigenvector $\vec{V}_0$ corresponding to the eigenvalue λ_0 is an approximation of $\vec{n} = \{n_x, n_y, n_z\}$ or -$\vec{n}$. The W_i term corresponds to the weighting of p_i and can be found as a function of the theoretical precision of its three-dimensional coordinates.

In the sequence, the k neighbors of $X$are searched. For each pair of points (X, p_k) three angles (α, ϕ, θ) that represent the local descriptor attributes obtained by the relation between each corresponding point to the normal $\vec{n}_X$ and $\vec{n}_k$are calculated. However, a LRS must be defined in X using the Equation (7). The Figure 4 shows the LRS and all attributes of the local descriptor.

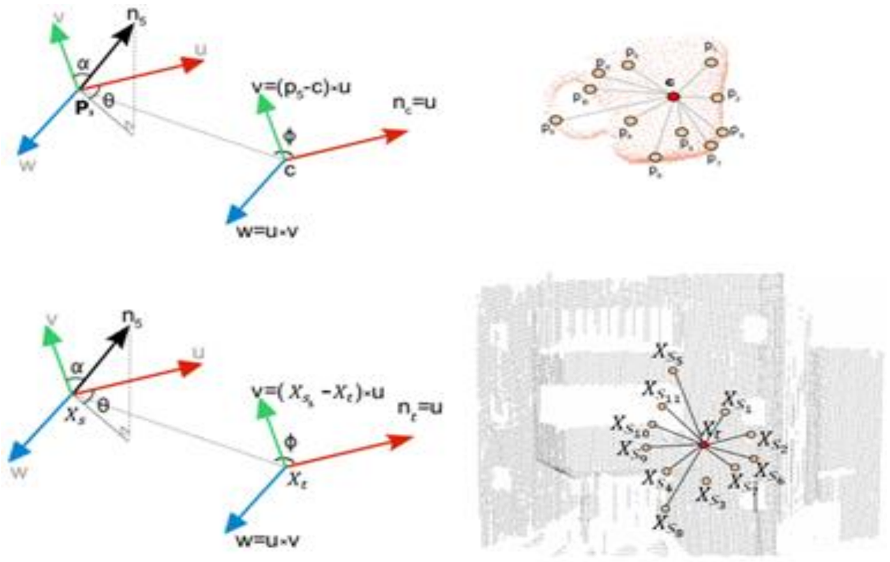

Figure 4: Attributes of the local descriptor and the LRS in X.

$$\begin{cases} u = \vec{n}_X \\ v = u \times \dfrac{(p_k - X)}{\|p_k - X\|^2} \\ w = u \times v \end{cases} \qquad (7)$$

The difference between the normal $\vec{n}_X$ and$\vec{n}_k$can be determined as follows:

$$\begin{gathered} \alpha = v \times \vec{n}_k \\ \phi = u \times \frac{(p_k - X)}{\|p_k - X\|^2} \\ \theta = \arctan(w \times \vec{n}_k, u \times \vec{n}_k) \end{gathered} \qquad (8)$$

The quadruple $<\alpha,\ \phi,\ \theta, \frac{(p_k-X)}{\|p_k-X\|^2}>$ is calculated for each pair of points $X$and p_k for sv_i, reducing from 12 attributes $(x^X, y^X, z^X, {n_x}^X, {n_y}^X, {n_z}^X, x^{pk}, y^{pk}, z^{pk}, {n_x}^{pk}, {n_y}^{pk}, {n_z}^{pk})$ to 4 attributes $\left(\alpha,\ \phi,\ \theta, \frac{(p_k-X)}{\|p_k-X\|^2}\right)$. To create the local descriptor of sv_i, the set of all $< \alpha_i,\ \phi_i,\ \theta_i>$ tuples is combined into a histogram. In this process, the distance measurements $\|p_k - X\|^2$ are divided into q bars of equal size creating a three-dimensional histogram with the total of q^3 points ($Hist$). Since the three attributes defined in the quadruple are measurements of angles between normal vectors, their values must be normalized to the same interval in a trigonometric circle. Figure 5 shows an example of a PFH histogram generated using 45 subdivisions to $\alpha,\ \phi, \theta$, plus 45 subdivisions of $\frac{(p_k-X)}{\|p_k-X\|^2}$ and 128 subdivisions for a component calculated as a function of the point of view which the angle histogram forms with each point normal, resulting in a vector of 308-byte values.

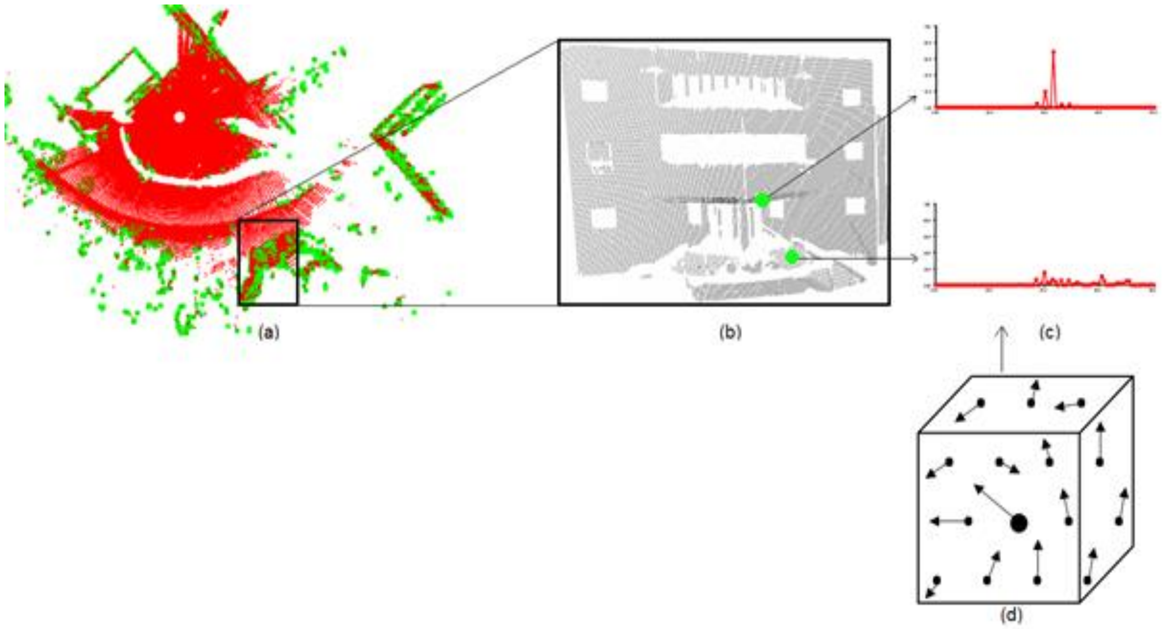

Figure 5: Example of PFH histogram. (a) 3D point cloud with several extreme points (b) highlighting of extreme points (c) histogram (d) voxel

In the 3D point cloud shown in Figure 5a, the extreme points in green and a selected region, presented in more detail in Figure 5b, with two histograms in Figure 5c, accounting for the number of occurrence of the each extreme point characteristics. On the other hand, Figure 5d, illustrates the orientations and magnitude of some extreme points. Given a set of histograms (local descriptors) in the filtered 3D reference point cloud $Hist^{M*}$ and a set of histograms in the filtered 3D search point cloud $Hist^{search*}$, the metric used ($d(Hist^{M*}, Hist^{search*})$) to establish the correspondence between the extreme points is given by:

$$d(Hist^{M*}, Hist^{search*}) = \frac{\sum_i \left(Hist^{M*}(i) - \overline{Hist^{M*}}(i)\right)\left(Hist^{search*}(i) - \overline{Hist^{search*}}(i)\right)}{\sqrt{\left(Hist^{M*}(i) - \overline{Hist^{M*}}(i)\right)^2 \left(Hist^{search*}(i) - \overline{Hist^{search*}}(i)\right)^2}} \quad (9)$$

where, $\overline{Hist} = \frac{1}{qq}\sum_j Hist(j)$, qq is the total number of histograms $Hist(j)$ in the reference or 3D search point cloud, respectively. Then, the Equation (09) is applied to every descriptor in order to find all pairs of corresponding descriptors. Consequently, the correspondences between the histograms are automatically established and an initial approximation of the 3D point cloud pair is performed using a 3D affine transformation.

The main advantage of the scheme developed in this step of the method is to avoid the iterative process by searching for matches using the proposed point-to-plane correspondence model. The initial approximations also prevent the proposed model from suffering of local minimums in the estimation of the rotation (R) and translation (t) parameters, in the step of pairwise 3D point cloud registration.

2.3. *Point-to-Plane Correspondence Model for Pairwise 3D Point Cloud Registration*

In this step of the method, a correspondence model based on a point-to-plane approach for pairwise 3D point clouds registration is proposed. The main characteristics of this model is the non-iterative correspondence step for estimating R and t. The proposed solution is based on the dissociation of the parameters of rotation (R) and translation (t) with lower computational cost. Firstly the rotation parameter (R) is calculated and then the translation parameter (t) is estimated without the need for iterations as done in the ICP algorithm.

Given a pair of point clouds $\aleph'$ (reference cloud) and $\beth'$ (search cloud), the RANSAC algorithm is used to extract planes in them. For this task, the algorithm randomly selects a minimum set of points (n) belonging to the data set $\aleph$ for generating candidate solutions and estimating the parameters of a model, where $\aleph \geq n$. Next, a description of the algorithm [2]:

a) Given a model that requires a minimum of n observations (subset S_1), with $\aleph \geq n$ for the estimation of a mathematical model M_1 from which a subset of observations S_1^* is determined, consisting of all points of $\aleph$ that have an error equal to or less than one tolerable error (e) pre-established. This group is known as Consensus and consists of 3D points called inliers. Those that remain above the tolerable error, are considered as outliers points;

b) If S_1^* is greater than a threshold τ (estimated as a function of the outliers points present in the set $\aleph$), a new mathematical model M_1^* will be determined, based on the application of the LSM. If the iteration (k) is not over, step (a) is returned;

c) If S_1^* it does not appear below the threshold τ, the algorithm randomly searches for a new subset (S_2), starting the process again.

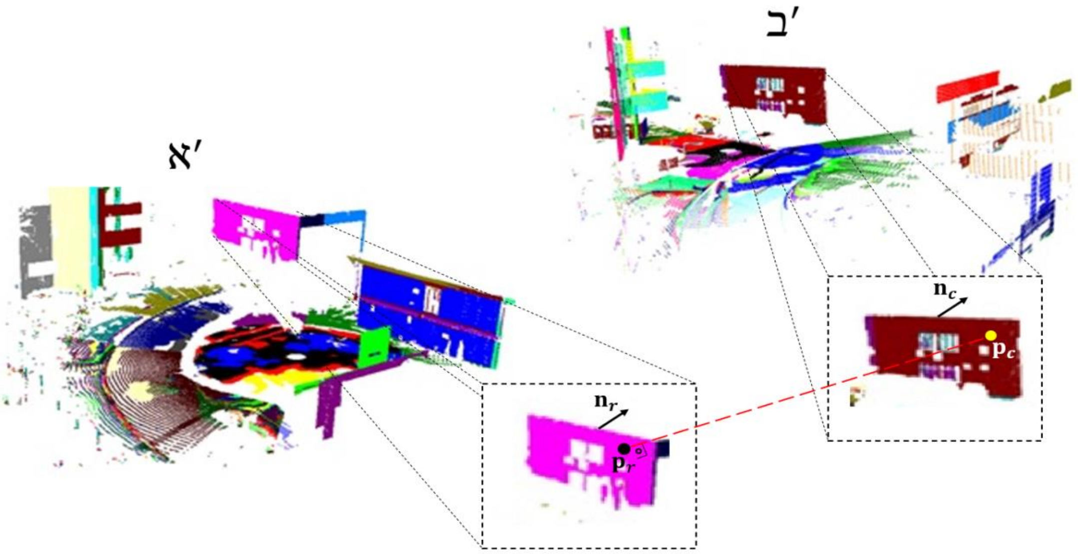

Figure 6: Conceptual basis of the proposed point-to-plan correspondence model.

Since initial approximations between $\aleph'$ and $\beth'$ were determined with SIFT3D + PFH, it can be assumed that for a given point p_r, belonging to the plane extracted π_R at $\aleph'$, its correspondent point $\mathrm{p}_c \in \beth'$ can be calculated through the intersection between the straight line, which is formed by the orthogonal projection of the point p_R and the plane $\pi_c \in ב'$, as shown in Figure 6.In practice, the corresponding point p_c and p_R is calculated as follows [11]:

$$n_c^T p_c = d_c$$

$$p_c = p_r + s n_c \qquad (10)$$

where s denotes a scalar, d_c is the distance from the origin of the referential system to the current plane at $ב'$ and n_c is the normal vector of the plane π_c at $ב'$.

To find the coordinates p_c, is necessarily just calculate the scalar s. This is done by replacing the first ($n_c^T p_c = d_c$) with the second term ($p_c = p_r + s n_c$) of Equation (11), as follows [11]:

$$\begin{cases} s = d_c - n_c^T p_r \\ p_c = p_r + (d_c - n_r^T p_r) n_c \end{cases} \qquad (11)$$

In this variation of the ICP algorithm, the correspondence between the primitives is established without the need for iterations. For each point belonging to the plane π_c at $ℵ'$, a correspondent point p_c^i (i = 1, ..., number of points in π_R) is calculated and a new normal vector ${n_c}^{new}$ to the plane $\pi_c \in ב'$ is estimated. To validate the establishment of point-to-plane correspondence, the angle between the vectors n_T and ${n_c}^{new}$ must be less than a pre-established threshold (ϟ), as follows:

$$\begin{cases} valido, se\ arccos\left(\frac{|n_T \cdot {n_c}^{novo}|}{\|n_T\| \|{n_c}^{novo}\|}\right) \leq \tau \\ invalido, caso\ contrário \end{cases} \qquad (12)$$

Now, whether $p_c = [x\ y\ z]^T \in ב'$, in the absence of systematic errors, the 3D rigid body transformation from point $p_r \in ℵ'$ to point p_c is given by:

$$p_c = R p_r + t \qquad (13)$$

In order to realize the pairwise 3D point cloud registration in this work, the following error function must be minimized:

$$e = \sum_i \|R p_r + t - p_c\|^2 \qquad (14)$$

Replacing the second term in Equation (11) in Equation (14):

$$e = \sum_i \|(R p_r + t) - [p_r + (d_c - n_r^T p_r) n_c]\|^2 \qquad (15)$$

Finally, the transformation parameters R and t are estimated using the Horn method [4]. In this method, the origin of the coordinate system of each 3D point cloud is translated to its centroid. The performance of this displacement allows the method to be realized in 2 stages, the first one consists of calculating the rotations, using quaternions, followed by the second, calculation of the translation. Next, the experiments and discussion of the results obtained using the method proposed in this work will be presented.

3. Experiments and Discussion of Results

As proof of concept of the method proposed in this work, LASER profiling of a terrestrial environment was performed using a TLS from the manufacturer FARO LS 800 with the following characteristics: 360° field of view in the horizontal direction and 120° in the vertical direction, maximum range between 15-400 meters, were generated for our purposes eight 3D point clouds (X_0, ...,X_7) with a density of 5 points/m², with overlap between the pairs of point clouds around 30% and the average distance between the TLS and the object of interest is around 20 meters. The area of interest encompasses the region of the Institute of Geosciences of the Federal University of Rio Grande do Sul in Brazil.

As described, the first step of the proposed method consists of detecting and removing outliers in point clouds, since the step of pairwise 3D point cloud registration is sensitive to the presence of these noises. In order to evaluate the outliers removal algorithm, experiments were carried out with each 3D point cloud using a value $\underline{d}$ = 50 cm and α = 0,10. The results can be seen in Table 1.

Table 1: Outlier removal results

Clouds	Nº of points (raw data)	Nº of points (After processing)
X_0	1,258,633	865,630
X_1	1,276,299	875,960
X_2	1,215,492	816,268
X_3	1,492,542	1,036,653
X_4	1,522,622	1,090,977
X_5	1,506,307	1,041,137
X_6	1,481,923	1,035,802
X_7	1,292,298	880,596

As can be seen in Table 1, before removing the outliers, the 3D point clouds had approximately 1,500,000 points. Using the outlier removal algorithm, around 31% of the points were discarded from the original 3D point cloud. This step of the method seeks, basically, to remove all points outside the range from the sample $\mu_k \pm \alpha \cdot \sigma_k$. In this work, the values assumed for the variables $\underline{d}$ and α were determined empirically, being those that best represented the expected sampling of the object on the surface. Figure 7 shows the 3D point clouds after the outlier removal process.

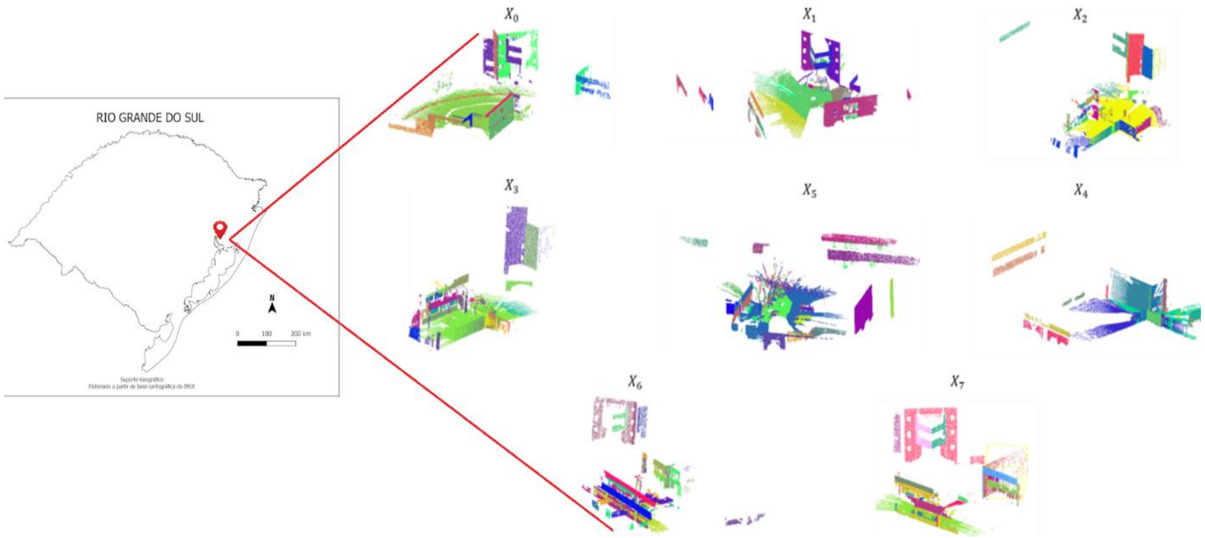

Figure 7: Point clouds remaining after using the outlier detection and removal algorithm.

The second step of the method proposed in this work consists of approximate initial values calculation between each pair of 3D point clouds. As previously described, this approach of estimating initial values is done by combining the SIFT3D algorithm with the PFH algorithm. For the performance of the SIFT3D algorithm, it is necessary to determine values of the threshold for the following variables: minimum scale, number of octaves and number of octaves per scale. The performance of the PFH algorithm is affected by the size of the neighborhood radius (sv_i). Figure 8 shows extreme points (green and blue points) detected in the 3D reference and search point cloud, respectively. As can be seen, the extreme points represent edge points and smooth or abrupt variations in the surface.

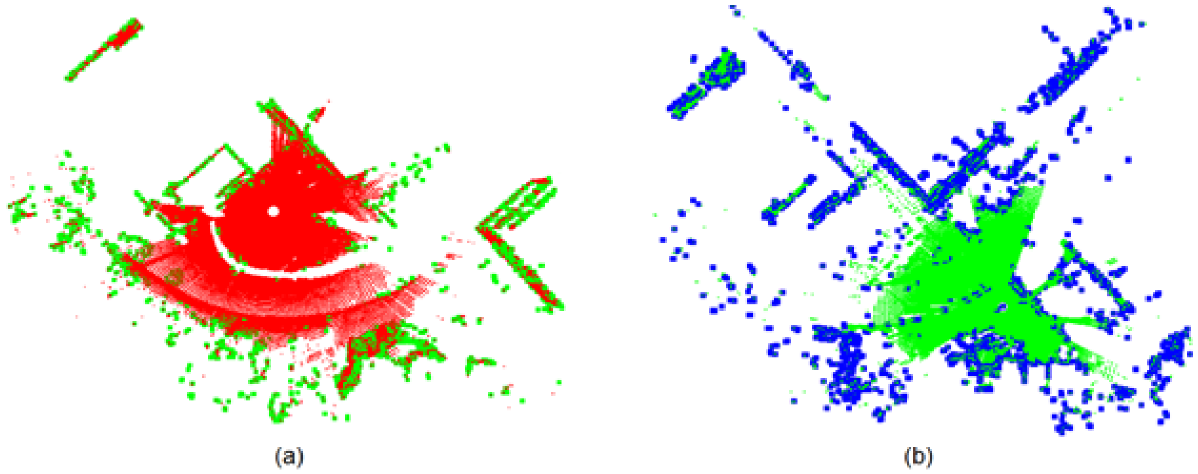

Figure 8: 3D extreme points: (a) 3D extreme points (in green) detected in the reference 3Dpoint cloud (red); (b) 3D extreme points (in blue) detected in the search 3D point cloud (in green)

Table 2 shows three experiments obtained with different threshold values for each of the aforementioned variables.

Table 2: Threshold values of the SIFT3D algorithm variables and average number of extreme points in each pair of point clouds

Minimum Scale	**Number of octaves**	**Number of octaves per scale**	sv_i	**Number of extreme points**	**Number of matches**
0.05	6	4	1.0 cm	3009	504
0.005	8	6	5.0 cm	1253	236
0.005	8	6	10.0 cm	67	52

The variables presented in Table 2, were determined empirically. These variables are expressly critical to the performance of the SIFT3D + PFH algorithm, since the calculation of the attributes of the local descriptor depends on the normal vectors of the extreme points, obtained by the SIFT3D algorithm. These normal vectors are determined as a function of the extreme point and its neighbors points contained in a radius circumference sv_1. The value of the variable sv_i is directly proportional to the amount of points and noise present in the 3D point cloud. However, lower the value of sv_1 better defined is the normal vector, thus avoiding the use of neighbor points unrepresentative the curvature of the surface where the extreme point belong. Consequently, the attributes of the local descriptor (calculated by the PFH) becoming better defined and represented by the resulting histogram. As a result, a greater number of true matches established arise leading to a better statistical quality. Another consequence is the reduction of the processing time to calculate the attributes of the descriptors, since the computational complexity will be $O(numero\_pontos\_vizinhos \cdot pontos\_extremos^2)$.

Figure 9 shows the results obtained with the proposed method for a pairwise initial approximation of the 3D point clouds. In Figure 9a, the 3D point clouds pairs are showed before the coarse alignment step. Figure 9b shows the results after the application of the coarse alignment to the 3D point cloud $ב'$ into the LRS of the reference cloud $א'$.

Table 3: Results of the segmentation of plans

3D Point Clouds	**Nº of planes**
X_0	50
X_1	70
X_2	59
X_3	58
X_4	40
X_5	90
X_6	70
X_7	60

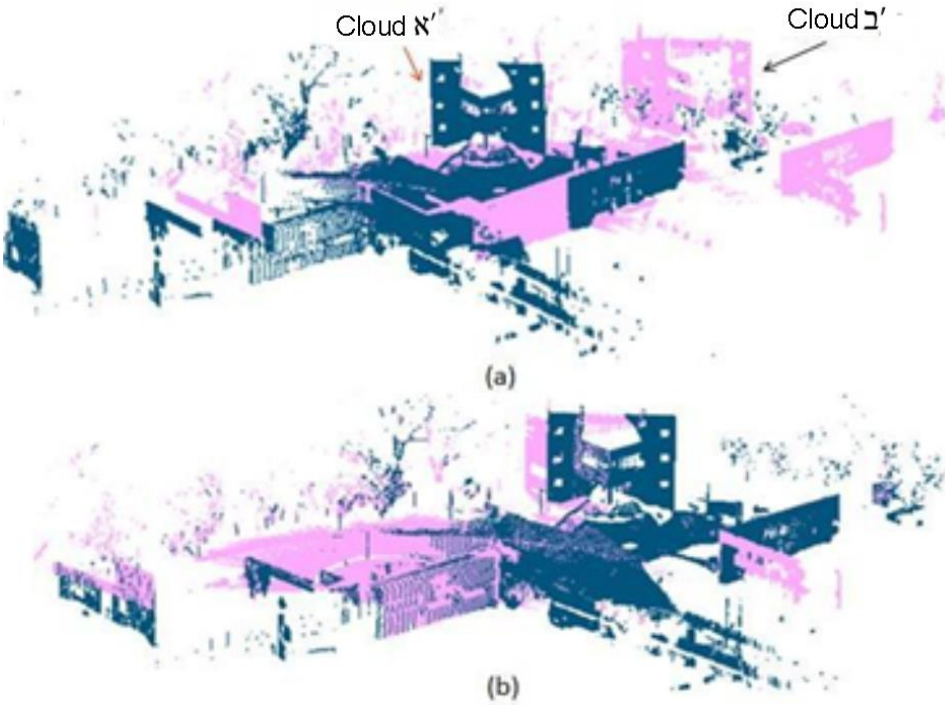

Figure 9: Result of the initial approximation between א′ and ב′. (a) Before and after (b) the initial alignment.

In this work, the coarse pairwise 3D point clouds registration provides robustness to the proposed point-to-plane correspondence model, reducing the abrupt variation of point of view displacement between the pair of 3D point clouds, since the overlap between them is only about 30%. This scheme of determining an initial transformation contributes to avoid the problem of mathematical model convergence in local maximums and minimums during the estimation of R and t.

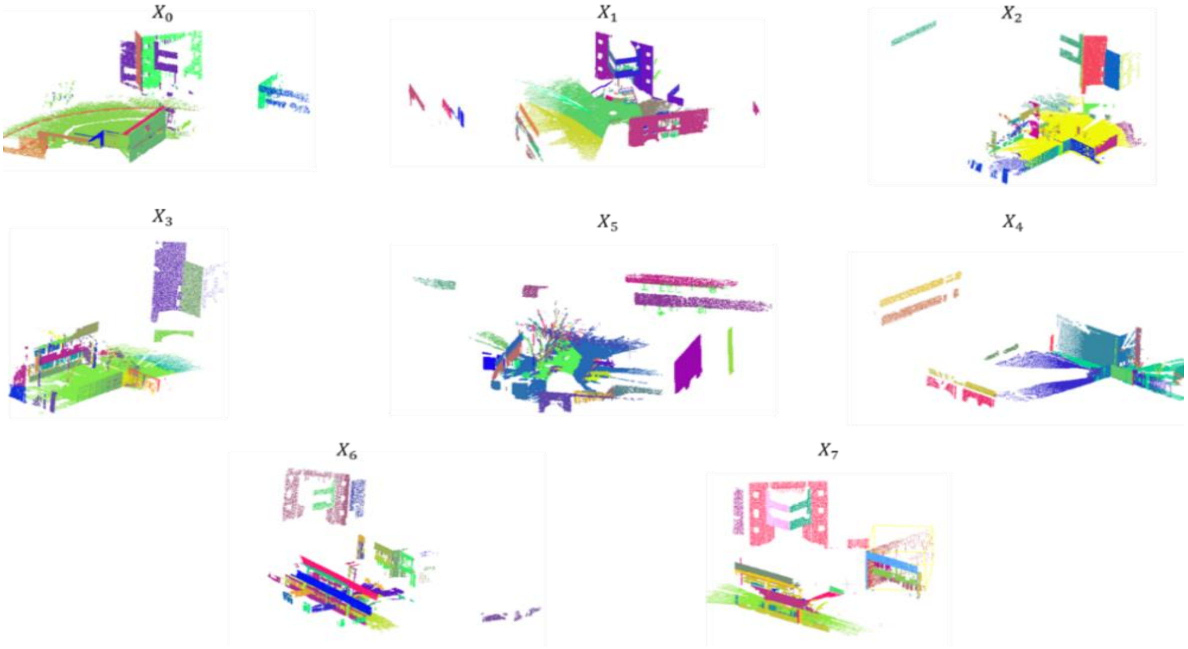

Figure 10: Segmented planes using the RANSAC algorithm

The estimation of the transformation parameters using the point-to-plane correspondence model is divided into three stages. Firstly, the plan segmentation process is performed using the RANSAC algorithm. In this work, points belonging to the plane were considered if has a distance less than 2.0 cm in relation to the analyzed plane. Table 3 shows the number of plans obtained with RANSAC and Figure 10 shows the segmented plans in each point cloud.

As can be seen in Table 3, the point cloud X_5 presented the largest number of segmented planes, however, visually it can be seen that most of the planes were extracted from regions with high density of vegetation, and they should be discarded from the matching process. It is also observed that, in all 3D point clouds, the plans referring to the ground were not segmented as a single plane. This can be explained due to the irregular topography of the profiled scenes.

Secondly, considering that a set of plans is extracted in $\aleph'$ (reference cloud) and their normal vectors are estimated with RANSAC $n_i = \left[n_x n_y n_z\right]^T$ and d_i, the point-to-plane model calculates each point orthogonally projected in $\beth'$ (research cloud) and estimates the normal vectors n_j and the distance from the origin to the plane d_j. The correspondence between the points of a plane in $\aleph'$ with the corresponding plane in $\beth'$ is correct if the angle between n_i and n_j is less than or equal to the angle threshold θ. In this work it was used θ = 0.5 °. The number of matching planes established using the proposed model is shown in Table 4.

Table 4: Number of matching plans

Pairs of 3D Point Clouds	Number of matching plans
$X_1 - X_0$	27
$X_2 - X_1$	38
$X_3 - X_2$	19
$X_4 - X_3$	11
$X_5 - X_4$	12
$X_6 - X_5$	16
$X_6 - X_7$	48
$X_7 - X_0$	39
$X_3 - X_0$	17
$X_2 - X_0$	22
$X_7 - X_5$	15

As can be seen in Table 4, for each pair of 3D point clouds, a number of matches were established greater than the degree of freedom required to apply the criteria for parameter optimization. Consequently, R and t are estimated using the point-to-plane correspondence model in a dissociative manner based on the method of Horn [4].

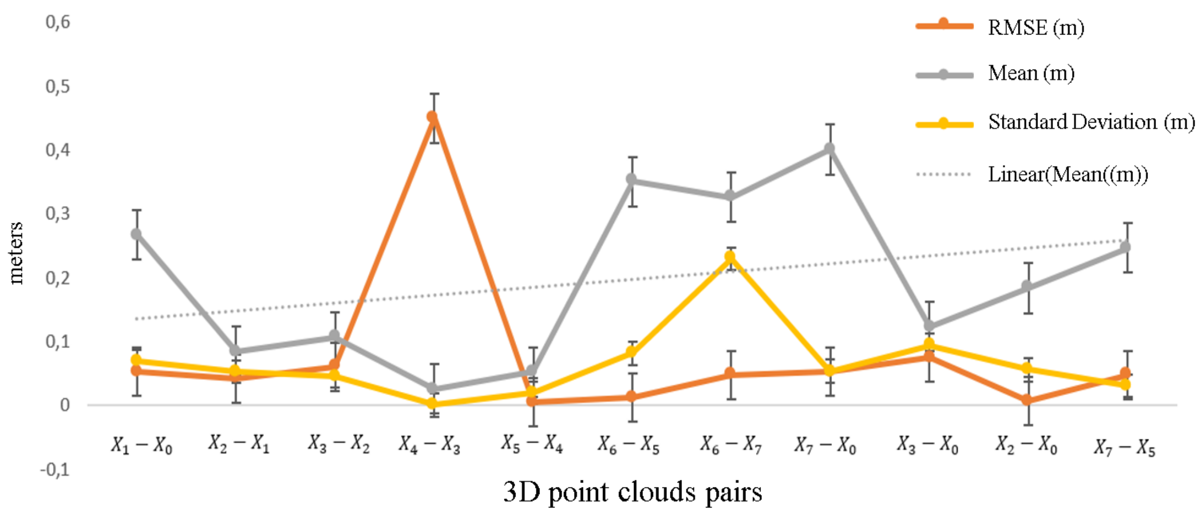

Figure 11: Results obtained with the propose method for the pairwise 3D point cloud registration

In order to statistically analyze the results obtained with the proposed method, the verification error was calculated. The verification error is represented by the absolute mean and the standard deviation of the distance between the centroid of the points belonging to a given plane in ב′ and its correspondent plane in the cloud א′, after R and t estimative. In order to verify the accuracy of R and t, the Root of the Mean Square Error (RMSE) of the planes distances residuals to the origin of the LRS of each pair of 3D point clouds was also calculated. Figure 11 shows these verification errors (mean and standard deviation).

The pairwise 3D point cloud registration $X_4 - X_3$, in Figure 11, produced the less convincing results, since the point-to-plane RMSE was about 0,45 m, due to the geometry of the planes and plane extraction process leading to ill-defined planes. Then in Figure 11, the values of the mean distance between the centroid of the points belonging to a given plane in ב′ and its correspondent plane in the 3D point cloud א′, after estimating R and t, show a systematic trend in the result (see straight line dotted in gray). This can be explained by the accumulation of systematic errors produced by the sensor during the data acquisition stage and by the insertion of random errors in the parameter estimation process. The accumulation of these errors causes closing error.

4. Conclusion

The coarse pairwise 3D point cloud registration is essential for the high performance of the propose method, since is the main source of misalignment between the registered 3D point clouds. In this work, the combination SIFT3D + PFH was able to provide approximations sufficiently adequate for the estimation of the transformation parameters. The correspondence model proposed to estimate the transformation parameters responded to expectations. The proposed method has the following advantages: (1) the plane matching is more stable than the correspondence between points; (2) plane surfaces are easily found in anthropic environments; (3) the planes are less influenced by noise; (4) it is robust to situations with low overlap between pairs of 3D point clouds; and (5) it does not depend on the sampling (or level of detail) of the data. As a recommendation for future work, it is suggested to apply parameter sensitivity tests.

References

[1] A. Gressin, C. Mallet, J. Damantké and N. David. “Towards 3D Deal Point Cloud Registration Improvement Using Optimal Neighborhood Knowledge” In ISPRS J. Photogram. Remote Sens, vol. 79, 2013, pp. 240–251.

[2] A. M. Fischler and C. R. 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, 1981.

[3] A. Nuchter, K. Lingemann and J. Hertzberg. “Cached KD Tree Search for ICP Algorithms”, In: Proc. Of The IEEE Conference 3-D Digital Imaging and Modeling, 2007, pp. 419-426.

[4] B. K. P. Horn. “Closed-Form Solution Of Absolute Orientation Using Unit Quaternions ”Journal Of The Optical Society Of America, vol. 4, pp. 629–642, 1987.

[5] C. Wei, T. Wu and H. Fu. “Plain-To-Plain Scan Registration Based On Geometric Distributions of Points” in IEEE International Conference on Information and Automation, ICIA 2015, 2015, pp. 1194–1199.

[6] D. Aiger, N. J. Mitra and D. Cohen-Or. “4-Points Congruent Sets for Robust Pairwise Surface Registration ”ACM Transactions on Graphics, vol. 27, pp. 1, 2008.

[7] D. Gibbins. 3D “Target Recognition Using 3-Dimensional Sift Or Curvature Keypoints And Local Spin Descriptors” Defense Applications Of Signal Processing. Kauai, 2009.

[8] D. Lowe. “Distinctive Image Features from Scale-Invariant Keypoints” In: International Journal Of Computer Vision, vol. 60, 2004, pp. 91-110.

[9] D. R. Dos Santos, M. A. Basso, K. Khoshelham, E. de Oliveira, N. L. Pavan and G. Vosselman. "Mapping Indoor Spaces by Adaptive Coarse-to-Fine Registration of RGB-D Data "in IEEE Geoscience and Remote Sensing Letters, vol. 13, 2016, pp. 262-266.

[10] D. W. Eggert, A. Lorusso A. and R. B. Fisher. “Estimating 3-D Rigid Body Transformations: A Comparison Of Four Major Algorithms ”Machine Vision And Applications, Springer Nature, vol. 9, pp. 272-290, 1997

[11] G. Dresch and Dos Santos D. R. “Automatic Assessment of Relative Accuracy of Data Dealing Airborne ”Geodetic Sciences Bulletin [Online], vol. 21, pp. 730–749, 2015.

[12] H. Li, D. Huang, J. M. Morvan, Y. Wang and L. Chen. “Towards 3D Face Recognition In The Real: A Registration-Free Approach Using Fine-Grained Matching Of 3D Keypoint Descriptors ”International Journal Of Computer Vision, vol. 113, pp. 128-142, 2015.

[13] K. H. Bae and D. D. Lichti. “A Method For Automated Registration Of Unorganized Point Clouds ”ISPRS Journal Of Photogrammetry And Remote Sensing, vol. 63, pp. 36–54, 2008.

[14] K. Khoshelham. “Closed-Form Solutions For Estimating A Rigid Motion From Plane Correspondences Extracted From Point Clouds ”ISPRS Journal Of Photogrammetry And Remote Sensing, vol. 114, pp. 78-91, 2016.

[15] K. S. Arun, T. S. Huang and S. D. Blostein. “Least Square Fitting Of Two 3-D Point Sets ”IEEE Trans. Patt. Anal. Intell machine. vol. Pami-9, pp. 698-700, 1987.

[16] N. L. Pavan and D. R. Dos Santos. “An Automatic Method for Registration Of Terrestrial Laser Scanning Data Using Planar Surfaces ”Geodetic Sciences Bulletin, pp. 572–589, 2015.

[17] P. Besl and N. Mckay. “A Method for Registration of 3-D Shapes” in IEEE Transactions On Pattern Analysis And Machine Intelligence, vol. 14, 1992, pp. 239–256.

[18] P. Henry, M. Krainin, E. Herbst, X. Ren and D. Fox. “RGB-D Mapping: Using Kinect-Style Depth Cameras For Dense 3D Modeling Of Indoor Environments ”The International Journal Of Robotics Research, vol. 31, pp. 647-663, 2012.

[19] P. W. Theiler, J. D. Wegner and K. Schindler. “Globally Consistent Registration of Terrestrial Laser Scans Via Graph Optimization ”ISPRS Journal Of Photogrammetry And Remote Sensing, vol. 109, pp. 126–138, 2015.

[20] R. B. Rusu, N. Blodow, M. Beetz. “Fast Point Feature Histograms (FPFH) For 3D Registration ”IEEE International Conference on Robotics And Automation, 2009, pp. 3212–3217.

[21] S. Rusinkiewicz and M. Levoy. “Efficient Variants Of The ICP Algorithm” in Proceedings Of Third International Conference On 3D Digital Imaging And Modeling. IEEE Computer Soc., 2001, pp. 145-152.

[22] T. Rabbani, S. Dijkman, V. D. Heuvel and G. Vosselman “An Integrated Approach For Modeling And Global Registration Of Point Clouds ”ISPRS Journal Of Photogrammetry And Remote Sensing. vol. 61, pp. 355-370, 2007.

[23] Y. Chen, and G. Medione. G. “Object Modeling by Registration Of Multiple Range Images. Image and Vision Computing” vol. 10, pp. 145–155, 1992.