*Article*

# Intensity-Assisted ICP for Fast Registration of 2D-LIDAR

**Yingzhong Tian** [1,2] **, Xining Liu** [1,2] **, Long Li** [1,2] **and Wenbin Wang** [3,*]

[1] School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200072, China; troytian@shu.edu.cn (Y.T.); liuxining@shu.edu.cn (X.L.); lil@shu.edu.cn (L.L.)
[2] Shanghai Key Laboratory of Intelligent Manufacturing and Robotics, Shanghai University, Shanghai 200072, China
[3] School of Mechanical and Electrical, Shenzhen Polytechnic, Shenzhen 518055, China
[*] Correspondence: wangwenbin@szpt.edu.cn

check for updates

**Abstract:** Iterative closest point (ICP) is a method commonly used to perform scan-matching and registration. To be a simple and robust algorithm, it is still computationally expensive, and it has been regarded as having a crucial challenge especially in a real-time application as used for the simultaneous localization and mapping (SLAM) problem. For these reasons, this paper presents a new method for the acceleration of ICP with an assisted intensity. Unlike the conventional ICP, this method is proposed to reduce the computational cost and avoid divergences. An initial transformation guess is computed with an assisted intensity for their relative rigid-body transformation. Moreover, a target function is proposed to determine the best initial transformation guess based on the statistic of their spatial distances and intensity residuals. Additionally, this method is also proposed to reduce the iteration number. The Anderson acceleration is utilized for increasing the iteration speed which has better ability than the Picard iteration procedure. The proposed algorithm is operated in real time with a single core central processing unit (CPU) thread. Hence, it is suitable for the robot which has limited computation resources. To validate the novelty, this proposed method is evaluated on the SEMANTIC3D.NET benchmark dataset. According to comparative results, the proposed method is declared as having better accuracy and robustness than the conventional ICP methods.

**Keywords:** iterative closest point; SLAM; best initial transformation guess; intensity residuals; anderson acceleration

## 1. Introduction

The estimation step carried out simultaneously to maintain the robot pose is regarded as the main work of simultaneous localization and mapping (SLAM) [1]. The iterative closest point (ICP) [2] is a well-known algorithm to deal with the laser scanner data presented in the form of point clouds. For example, an improved probabilistically-motivated maximum likelihood estimation (MLE) algorithm including the ICP algorithm was used in [3] to adaptively decrease the search scope in an efficient real-time mobile unmanned ground vehicle (UGV) indoor positioning system for large-area applications. A vehicle localization method based on a vertical corner feature used ICP algorithm to deal with the geometric relations between the scan data of the 3D light detection and ranging (LIDAR) [4]. A technique to estimate the ego motion of a vehicle was presented in [5] where the ICP algorithms were evaluated for their accuracy and computational speed based on laser range data. A 3D mapping system proposed an art ICP-based SLAM method [6] using only point cloud data. A robust and accurate SLAM algorithm for omnidirectional mobile robots based on a novel 2.5D LIDAR device and ICP algorithm was presented in [7,8]. In their works, the 2.5D LIDAR device could

also perform vertical scanning within the motion range of the liner stage. Hence, the 2.5D points cloud could be performed by the ICP algorithm to improve the robustness and accuracy of the scan matching. Besides the LIDAR SLAM, the ICP algorithm is also used in the problem of scan-matching in visual SLAM. An adaptation of the ICP algorithm was introduced for the matching of straight lines between two stereo image pairs so that the speed improvement and local minima reduction can be seen in visual SLAM [9]. A real-time, robust, low-drift and depth-only SLAM system applied a local map-based motion estimation strategy using the ICP algorithm to reduce the local drift [10].

The concept of the ICP is straightforward. Initially, it estimates the corresponding points between the source and target frame by utilizing a certain metric. Then it estimates the relative transformation with a closed-form solution which aims to minimize the distance between the correspondences. These two steps are iteratively performed until a convergence criterion, or the maximum number, is precisely reached.

There have been many modified ICP algorithms presented nowadays. Globally optimal iterative closest point (Go-ICP) was introduced in [11] to speed up the conventional ICP. Point-to-line metric iterative closest point (PLICP) could converge in a finite number of steps with quadratic speed [12]. A tensor-based ICP did some works on the optimal initial transformation guess [13]. A 3-step approach based on ICP and point Cloud Projection (ICP-CP) performed well for both enhancing the accuracy and reducing the execution in most cases [14]. 4D-ICP was put forward in [15] to accelerate the registration of 3D point cloud segments with hue data from the associated imagery. Besides that, there were lots of variant ICP such as [16–19].

Intensity information extracted from LIDAR data or visual data sometimes can be used for registration and localization. An intensity-based medical image registration added intensity information into ICP algorithm for scan-matching [20]. The paper [21] put forward a new ICP algorithm assisted with intensity information for visual ranging estimation from a RGB-D camera in visual SLAM. An automatic registration system was presented for aligning intensity scan pairs which was designed to handle several challenges including extensive structural changes, large viewpoint differences, repetitive structure, illumination differences and flat regions [22]. A robust vehicle localization method applied a novel fusion algorithm assisted with vertical and road intensity information for robust localization based on a prior point cloud in urban area [23].

Slightly different from [21] mentioned above, this paper presents a fast and robust ICP based on the assisted intensity method using both the intensity and distance. In addition to adopting the Anderson method encompassed under the used acceleration algorithm, this paper contains the following contributions.

The proposed method utilizing the assisted intensity approach to compute the initial transformation guess of ICP different from that of [21]. The intensity information in [21] is used to be set up as robust weighting function to correct the next transformation in the iterative process with an initial transformation guess still set up as identity matrix. However, the intensity information in this paper is added to a target function to determine the best initial transformation guess based on the statistics of spatial distances and intensity residuals of the point cloud with an iteration process being no robust weighting function.

The rest of this paper is organized as follows. Section 2 describes the conventional ICP method [2] and the Anderson acceleration [24]. The proposed intensity-assisted ICP method is explained in Section 3. Section 4 presents the performance of the proposed method depicted by the evaluation of the SEMANTIC3D.NET benchmark dataset. Finally, a conclusion is presented in Section 5.

## 2. Preliminaries

### 2.1. Conventional Iterative Closest Point (ICP)

The ICP algorithm is commonly used for the robot navigation in performing scan-matching of certain data generated by LIDAR. The source frame $P_1$ assisted by intensity $I_1$ can be consider as

$<P_1, I_1>$. Then the target frame is the same as the source frame aligned as $<P_2, I_2>$. Furthermore the conventional ICP algorithm can be described as follows:

(1) Transform $P_1$ using initial transformation guess $T_0$ which is usually set as an identity matrix.

(2) Search for each point in $P_1$ a closet point in $P_2$ as correspondence with the transformation $T_{k-1}$ in the $k$-th iteration of algorithm. For the $i$-th point in $P_1$, the index of corresponding point in $P_2$ is described as *index*($i$), where:

$$index(i) = \underset{j}{\operatorname{argmin}} \| T_{k-1} P_1(i) - P_2(j) \|. \tag{1}$$

(3) Discover transformation $T_k$ that minimizes the distance between the correspondence:

$$T_k = \underset{T}{\operatorname{argmin}} \sum_i \| T T_{k-1} P_1(i) - P_2(index(i)) \|. \tag{2}$$

(4) Apply transformation $T_k$ to $P_1$.

(5) Stop algorithm when the incremental transformation is smaller than the threshold or oppositely proceed to Step 2.

*2.2. Anderson Acceleration*

The Anderson acceleration [25,26] is a recursive procedure that is usually used to find fixed points of contractive mapping that have been considered to have better performance than conventional picard iteration procedures. Pavlov et al. [24] proposed Anderson accelerated version of ICP (AA-ICP) by utilizing Anderson acceleration. Similarly, Anderson acceleration is also used in this paper to increase the iteration speed to quickly get the final transformation. In the $k$-th iteration, the Anderson acceleration can be described as follows.

(1) As mentioned above in Section 2.1, the process of transformation $T_k$ derived from last transformation $T_{k-1}$ can be described as a contraction mapping $G$. Thus $T_k$ be described by referring to the following Equation (3):

$$T_k = G(T_{k-1}). \tag{3}$$

(2) The error between current transformation $T_k$ and last transformation $T_{k-1}$ can be described as $f_{k-1}$. Hence $f_{k-1}$ can be obtained from Equation (4) below:

$$f_{k-1} = T_k - T_{k-1}. \tag{4}$$

(3) The goal of the algorithm is to find $\alpha$ which minimizes,

$$\sum_{j=1}^{n} \alpha_j f_j, \tag{5}$$

where,

$$\sum_{j} \alpha_j = 1. \tag{6}$$

(4) The optimal transformation $T_{k+1}$ can be produced by the following Equation (7):

$$T_{k+1} = \sum_{j=1}^{n} \alpha_j G(T_j). \tag{7}$$

(5) Apply transformation $T_{k+1}$ into source frame aiming to measure the distance between two frames. Furthermore, the Anderson acceleration will be stopped if the incremental iteration is smaller than a threshold or, by contrast, will proceed to Step 1.

The core content of Anderson acceleration is to minimize iterations as little as simple iterations to blend with the same error. The iteration step cost is neglected compared to a single ICP iteration step while it depends on iteration history. The most important thing is that it can be added trivially to the existing ICP implementation with little modification.

## 3. Intensity-Assisted Iterative Closest Point

The intensity-assisted ICP proposed in this paper is different from the conventional ICP as described below.

(1) **Initial Transformation Guess**—The initial transformation guess is not set as an identity matrix, and determined by utilizing the proposed target function which is based on the statistics of their spatial distances (see detail in Section 2.1) and intensity residuals. It is computed with assisted intensity for their relative rigid-body transformation which aims to reduce the computational cost and avoid divergence.

(2) **Anderson Acceleration**—Anderson acceleration is different from the standard Picard iteration which is commonly used in the conventional ICP. It is used to increase the speed of iteration for quickly reaching a convergence (see details in [24]).

An overview of the proposed ICP method compared with conventional ICP in this paper can be seen in Figure 1.
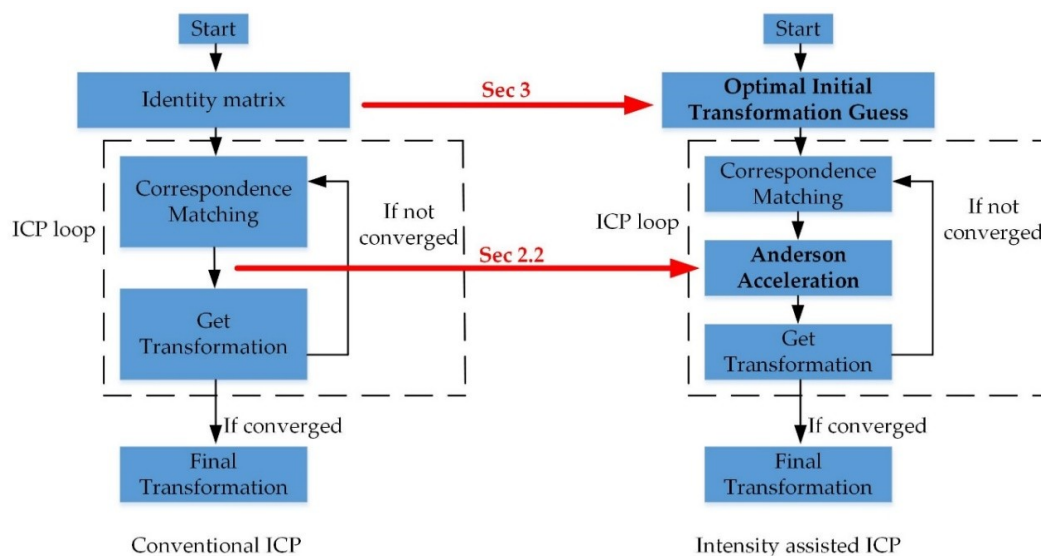


**Figure 1.** An overview of intensity-assisted iterative closest point (ICP) method compared with conventional ICP.

As can be observed in the conventional ICP, the initial transformation guess is usually set as identity matrix. This may lead to divergence when the distance of the source and target frame is too large. For this reason, the intensity residuals between correspondences are considered to obtain the optimal initial transformation guess as described below.

### 3.1. Salient Intensity Point Selection

The cost of the computation for the initial transformation may be enormous if there is no sampling process for the point cloud. A new sampling method is proposed in this paper which aims to select the salient point assisted with the intensity that is different from the random sampling method in the point cloud library (PCL).

In this paper, a market square dataset called marketsquarefeldkirch4-reduced that can be found from the SEMANTIC3D.NET benchmark dataset is used in the proposed method. In this dataset, a 2D point clouds of the ground shown in Figure 2 are extracted to support the experiment. The intensity

value of the method used to select the salient point is obtained from this dataset. Due to the intensity information in this dataset extracted from 3D Velodyne LIDAR, the method in this paper can perform well in other LIDAR datasets. However, this method may be not suitable for visual point cloud due to the different intensity structure between the LIDAR point cloud and visual point cloud.
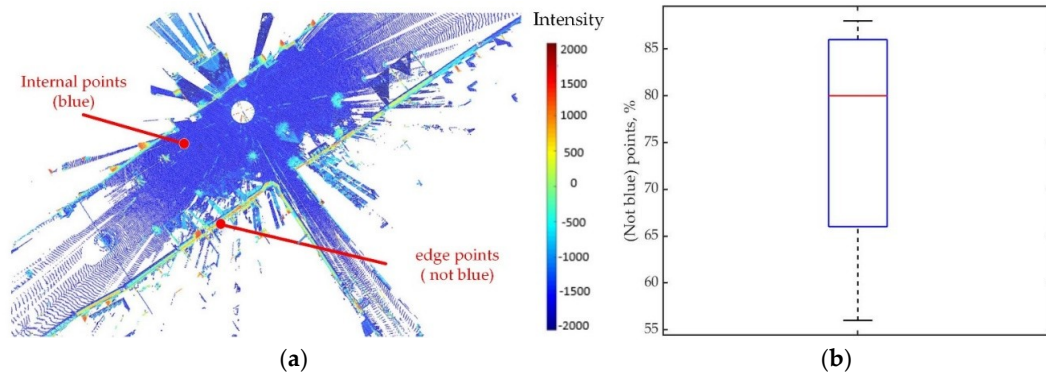


(**a**)                    (**b**)

**Figure 2.** (**a**) Depending on the distribution of intensity value in point cloud, it can be found that the high-intensity points (not blue) are usually on the edge of point cloud with low-intensity points (blue) inside. Therefore, the point cloud composed of high intensity points can represent the shape of the point cloud. (**b**) Based on 100 times experiments to intensity point selection, it can be seen that most of points (not blue) can be extracted for filtered point cloud by Equations (8) and (9).

As shown in Figure 2a, the intensity value of most of points on the edge of the point cloud is higher than −1000. This is caused by the most points on edge are green points with the intensity of between −1000 and 0. Therefore, the criterion of the salient point $x$ is:

$$-1000 < I_1(x) < 0. \tag{8}$$

$$-1000 < I_2(x) < 0. \tag{9}$$

The filtered point cloud is produced by the random sampling method with this criterion. As can be seen in Figure 2b, the threshold about the −1000 and 0 can be used to obtain 80% of the edge points for filtered point cloud by 100 times the random sampling method. This can prove the validity of this threshold.

Besides that, the number of the selected points is set to 100 reference to [21] so that it can be used to reduce the cost of computation but not affect the accuracy of the experimental results. Thus, the point cloud filtered by the new sampling method presented in this paper is depicted in Figure 3.
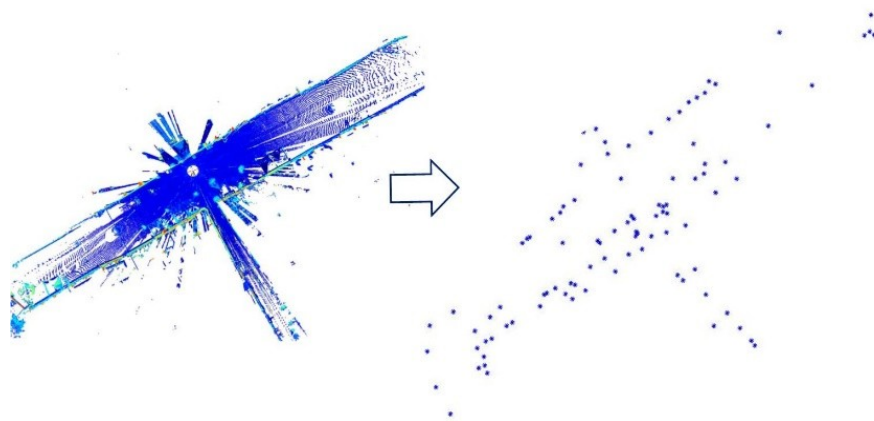


**Figure 3.** It can be seen that some high intensity points on the edge of the point cloud can be extracted for next registration without so much cost of computation using the method in Section 3.1.

As can be seen, the selected points are usually located on the edge of the point cloud with the internal points having disappeared. By using sampling method mentioned above, the filtered point cloud can represent the direction and shape of the point cloud. Then the filtered point cloud is set up as the source frame and transformed to the target frame with a certain number of candidates of initial transformation guess. Once the filtered point cloud is chosen, it is used for all candidates of initial transformation guess. When the optimal initial transformation guess is obtained, the point cloud for later iteration is the original point cloud but not the filtered point cloud.

### 3.2. Discover Optimal Initial Transformation Guess

In this step, a certain amount of initial transformation guesses $\{u_1, u_2\ u_3, u_4, \ldots, u_n\}$ will be produced by the algorithm. Then, the correspondences between filtered source frame and filtered target frame is established by each initial transformation guess $u_i$.

Despite the sampling method to choose the points randomly, it can generally cover the entire edge of the point cloud with the criterion in experiment as described in Figure 3. With that the filtered source frame and filtered target frame have the same direction and shape. Hence, a proper transformation can be obtained.

A target function is proposed to determine the best initial transformation guess $u_{optimal}$ based on statistics of their spatial distances and intensity residuals. Intensity information can be demonstrated so that it can be used to registration and localization as mentioned in [20,22,23].

For example, the corresponding point of $i$-th point in source frame $P_1(i)$ is $P_2(index(i))$ with an initial transformation guess $u_i$. The intensity residuals and spatial distances between the correspondences are considered for the target function used in this experiment. The intensity residuals $r_i^{(I)}$ of the $i$-th pair are calculated as:

$$r_i^{(I)} = I_1(i) - I_2(index(i)). \tag{10}$$

Inspired by [21,27], the Student's $t$-distribution is used to scale each intensity residual. The conventional method uses the $t$-distribution to reduce the outliner influence and to determine the correspondence correctly in [21]. By contrast, the use of $t$-distribution in this paper focuses on the computation of its variance which is obviously different from the conventional method. By referring [28], the intensity component of the target function is derived based on $t$-distribution. Thus, weight function $w_I(r_i^{(I)})$ is:

$$w_I(r_i^{(I)}) = \frac{v_{(I)} + 1}{v_{(I)} + \left(\frac{r_i^{(I)}}{\sigma^{(I)}}\right)^2}. \tag{11}$$

In this experiment, the degree of freedom (DoF) $v_{(I)}$ is set to 2.055 and the initial standard variance $\sigma^{(I)}$ is set to 7.7189. Notice that the $v_{(I)}$ and $\sigma^{(I)}$ come from the "fitdist" function in MATLAB. Although $v_{(I)}$ and $\sigma^{(I)}$ are based on current dataset, the value can still be applied to other Lidar datasets (see Table 1).

By contrast with the usage in [27], the variance of $t$-distribution used in this experiment is used to determine the best initial transformation guess. When a correspondence between the filtered source frame and filtered target frame is established by an initial transformation guess, the $score^{(I)}$ is computed by using Equation (12):

$$score^{(I)} = \frac{1}{n}\sum_i (r_i^{(I)})^2 w_I(r_i^{(I)}). \tag{12}$$

The $score^{(I)}$ is the same as the variance of $t$-distribution. The effect of this score is illustrated in Figure 4a where the variance of optimal $t$-distribution produced by $u_{optimal}$ is lower than the initial $t$-distribution produced by $u_1$. It means that most of values of the intensity residuals are zero when optimal initial transformation guess is given. The optimal initial transformation guess can be found in the search process when the $score^{(I)}$ reaches minimum.
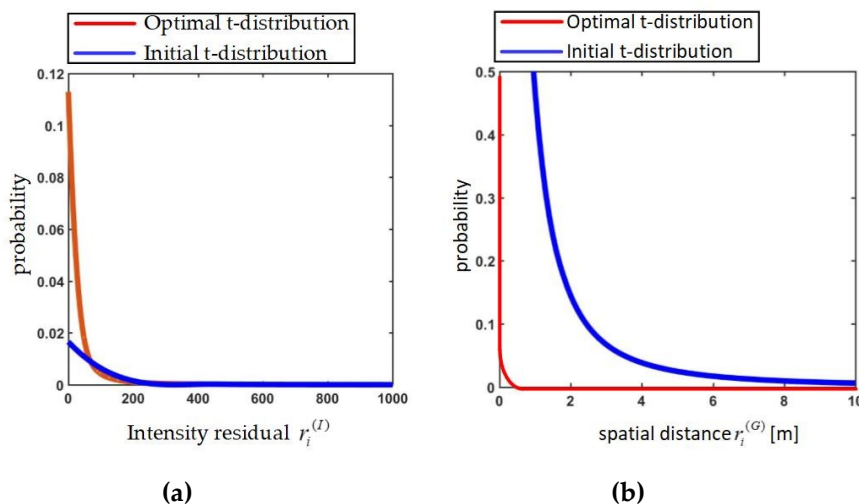
**(a)**　　　　　　　　　　　　　**(b)**

**Figure 4.** (**a**) The variance of optimal t-distribution about the intensity residuals is lower than that given by initial t-distribution. (**b**) The variance of optimal t-distribution about the spatial distances is lower than that given by initial t-distribution.

Another component of the target function is based on the spatial distances between correspondences. The spatial distance $r_i^{(G)}$ is considered as follows:

$$r_i^{(G)} = \|T_{k-1}P_1(i) - P_2(index(i))\|. \tag{13}$$

The computation of $w_G(r_i^{(G)})$ follows the same method for the intensity residual.

$$w_G(r_i^{(G)}) = \frac{v_{(G)} + 1}{v_{(G)} + \left(\frac{r_i^{(G)}}{\sigma^{(G)}}\right)^2}. \tag{14}$$

The degrees of freedom (DoF) of the spatial distances $v_{(G)}$ is set to 1.22 which is different from $v_{(I)}$. The initial standard variance $\sigma^{(G)}$ is set to 0.5326. Then, the $score^{(G)}$ can be calculated by Equation (15):

$$score^{(G)} = \frac{1}{n}\sum_i (r_i^{(G)})^2 w_G(r_i^{(G)}). \tag{15}$$

The effect of the $score^{(G)}$ can be described the same as $score^{(I)}$. When the optimal *t*-distribution of spatial distance is obtained by utilizing optimal initial transformation guess, the variance is close to zero as can be seen in Figure 4b.

Referring to Figure 4a,b, the variance of t-distribution fits the real data significantly. Then the $score^{(I)}$ and $score^{(G)}$ are all normalized for the same weight by Equation (16).

$$SCORE_{u_i}^{(I)} = \frac{score_{u_i}^{(I)}}{\sum_{n=1}^{n} score_{u_i}^{(I)}}, SCORE_{u_i}^{(G)} = \frac{score_{u_i}^{(G)}}{\sum_{n=1}^{n} score_{u_i}^{(G)}}. \tag{16}$$

Thus, the curves of the $SCORE^{(I)}$ and $SCORE^{(G)}$ in the search process can be depicted by Figure 5a,b, respectively. The $SCORE^{(I)}$ will be extremely close to the minimum the same as $SCORE^{(G)}$ when optimal initial transformation guess is given. However, $SCORE^{(I)}$ and $SCORE^{(G)}$ may not necessarily sync to the minimum which means that the $SCORE^{(I)}$ has multiple minimums. The reason is that the correspondence may be identified when a part of the initial transformation guess is close to the given optimal value. Thus the $SCORE^{(I)}$ computed by optimal initial transformation guess may be the same as others around the optimal value.
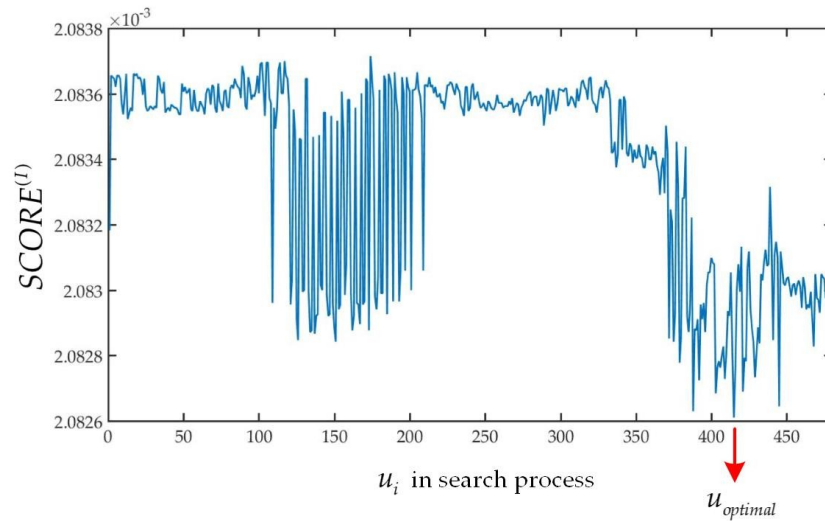
(**a**)
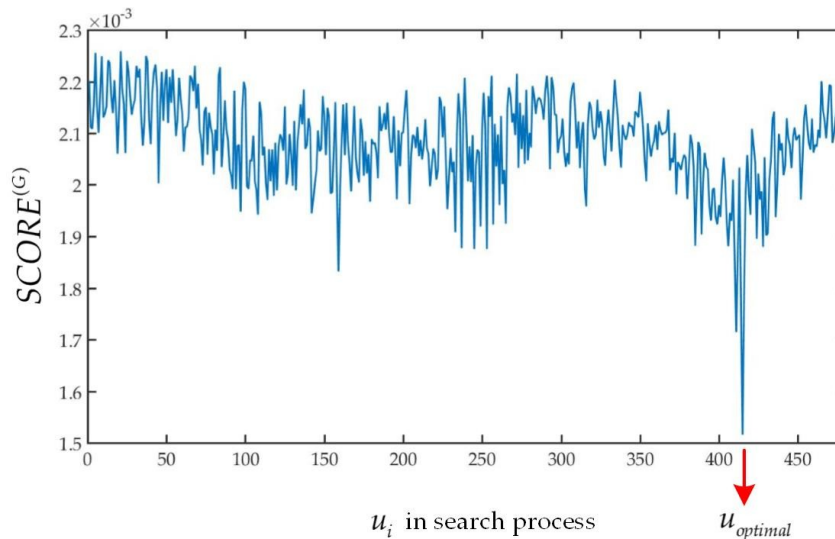


(**b**)

**Figure 5.** (**a**) The $SCORE^{(I)}$ curve will reach the minimum when optimal initial transformation guess is used. (**b**) The $SCORE^{(G)}$ curve will reach the minimum when optimal initial transformation guess is used.

However, the $SCORE^{(I)}$ can help to find the optimal initial transformation guess due to it being able to access to the area including the optimal value. Then the corresponding $SCORE^{(G)}$ for the same $SCORE^{(I)}$ are different. By combined with $SCORE^{(G)}$, the target function that used to find the optimal initial transformation guess $u_{optimal}$ is described as follows:

$$u_{optimal} = \underset{u_1, u_2 \dots u_n}{\operatorname{argmin}} \left( SCORE_{u_i}^{(I)} SCORE_{u_i}^{(G)} \right). \tag{17}$$

With the new method which is assisted by the intensity, the optimal initial transformation guess can be quickly obtained so that the proceeding speed of ICP will be increased.

Besides the method to compute the initial transformation guess, the Anderson acceleration is also used to increase the speed of iteration. Therefore, the entire algorithm of ICP assisted by the intensity method is represented by Algorithm 1.

---

**Algorithm 1: Intensity-Assisted ICP**

---

**Input:** Source frame $<P_1, I_1>$ and target frame $<P_2, I_2>$, the search criteria $C_\theta, C_x, C_y$, the search step $S_\theta, S_x, S_y$

**Output:** Convergence transformation $T_{final}$

get sampled point cloud from $P_1, P_2$ with Section 3.1

get $C_\theta, C_x, C_y, S_\theta, S_x, S_y$ with Algorithm 2

$n = 1$

**for** $\theta = -C_\theta: S_\theta: C_\theta$ **do**

**for** $x = -C_x: S_x: C_x$ **do**

**for** $y = -C_y: S_y: C_y$ **do**

get the $SCORE^{(I)}$ and $SCORE^{(G)}$ with $u_n$ produced by $\theta, x, y$

$n++$

**end for**

**end for**

**end for**

get $u_{optimal}$ with Section 3.2

$T_0 = u_{optimal}$

Anderson acceleration begin with $T_0$ between $P_1, P_2$

when convergence criteria is true, return $T_{final}$

---

In this algorithm, it is set that $C_\theta = 45°$, $C_x = 1$, $C_y = 1$, $S_\theta = 3$, $S_x = 0.2$, $S_y = 0.2$ for large distances registration with $u_{max} = u_{3000}$ or $C_\theta = 10°$, $C_x = 0.2$, $C_y = 0.2$, $S_\theta = 1$, $S_x = 0.04$, $S_y = 0.04$ for small distance registration with $u_{max} = u_{2000}$. An adaptive algorithm is introduced in this paper to select the thresholds above automatically so that the intensity-assisted ICP can perform well in continuous registration. The process to choose the thresholds is shown below.

In this step, the filtered point clouds obtained in Section 3.1 are used for adaptive selection of thresholds. Inspired by [16], the mean $\mu$ and the covariance $\Sigma$ of filtered point clouds $P$ is produced through Gaussian distribution as follows:

$$\mu = \frac{1}{|P|} \sum P(i), \quad \sum = \frac{1}{|P|} \sum (P(i) - \mu)^T (P(i) - \mu). \tag{18}$$

Then the covariance $\Sigma$ is used to obtain eigenvector $\xi_1, \xi_2$ by eigenvalue decomposition as follows.

$$\sum = (\xi_N, \xi_L) \begin{pmatrix} \lambda_N & 0 \\ 0 & \lambda_L \end{pmatrix} (\xi_N, \xi_L)^T. \tag{19}$$

Here $\lambda_N$ and $\lambda_L$ are the eigenvalues of $\Sigma$ in ascending order, and $\xi_N, \xi_L$ can represent the axes of the ellipsoid approximating the point distribution. The eigenvector $\xi_N$ corresponding to the smaller eigenvalue of $\xi_N, \xi_L$ is the normal vector so that it describes the direction of the point cloud excellently.

Finally threshold selection is done from the filtered source frame $P_1$ and the filtered target frame $P_2$ using Equations (20) and (21):

$$\|\mu_1 - \mu_2\| \leq 0.2. \tag{20}$$

$$\|\xi_{1,N} - \xi_{2,N}\| \leq 10°. \tag{21}$$

When Equation (20) or Equation (21) is satisfied in algorithm, it is set that $C_\theta = 10°$, $C_x = 0.2$, $C_y = 0.2$ and $S_\theta = 1$, $S_x = 0.04$, $S_y = 0.04$. If not, then the parameters can be set to $C_\theta = 45°$, $C_x = 1$, $C_y = 1$ and $S_\theta = 3$, $S_x = 0.2$, $S_y = 0.2$. The adaptive algorithm to select thresholds is described by Algorithm 2 and Figure 6.

---

**Algorithm 2: Adaptive Threshold Selection**

---

**Input:** The filtered source frame $P_1$ and the filtered target frame $P_2$
**Output**: The search criteria $C_\theta$, $C_x$, $C_y$, the search step $S_\theta$, $S_x$, $S_y$
get $\mu_1$, $\mu_2$, $\xi_{1,N}$, $\xi_{2,N}$ with Equations (18) and (19)
**if** satisfy with Equation (20) || satisfy with Equation (21) **then**
$C_\theta = 10°$, $C_x = 0.2$, $C_y = 0.2$ and $S_\theta = 1$, $S_x = 0.04$, $S_y = 0.04$
**else**
$C_\theta = 45°$, $C_x = 1$, $C_y = 1$ and $S_\theta = 3$, $S_x = 0.2$, $S_y = 0.2$
**end if**
**return** the $C_\theta$, $C_x$, $C_y$ and $S_\theta$, $S_x$, $S_y$
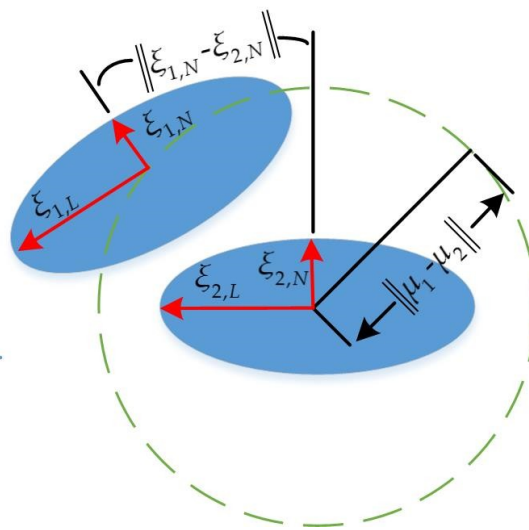
---



**Figure 6.** It can be seen that a normal vector can excellently describe the direction of point cloud.

## 4. Experimental Results

The proposed intensity assisted ICP algorithm is evaluated using a new large-scale point cloud classification benchmark called SEMANTIC3D.NET [29]. This data set means as for data-hungry (deep) learning methods with over four billion manually labelled points. The SEMANTIC3D.NET benchmark dataset consists of dense point clouds acquired with static terrestrial laser scanners. It contains eight semantic classes and covers a wide range of urban outdoor scenes: churches, streets, railroad tracks, squares, villages, soccer fields and castles, providing more dense and complete point clouds with a much higher overall number of labelled points compared to those already available to the research community.

Due to our new ICP method containing the intensity information, the SEMANTIC3D.NET benchmark dataset is suitable for our experiment compared to other data sets.

In this paper, the proposed method focus on 2D datasets produced from the SEMANTIC3D.NET benchmark dataset, however, it may perform well on 3D ICP as well.

To validate the performance of the proposed method, it was compared with the conventional ICP in point cloud library (PCL). This comparison was initiated by performing both mentioned methods on a desktop computer with Ubuntu 16.04, equipped with Intel Core i7-6700HQ CPU (2.6 GHz) and 16GB RAM. Note that this implementation was only run on a single CPU thread.

All subsequent experiments are based on a dataset called marketsquarefeldkirch4-reduced in the semantic-8 benchmark (contains approximately 10,538,633 points). The part of this point cloud is selected where $z$ is less than 2 m and also mapped to the $x$-$y$ plane for the experiments.

The iteration examples of the intensity-assisted ICP and conventional ICP in PCL are shown in Figures 7 and 8. Based on these figures, it can be shown that the properties of the accelerative statistics for the number of iterations required for convergence is calculated for $\varepsilon = 0.001$.
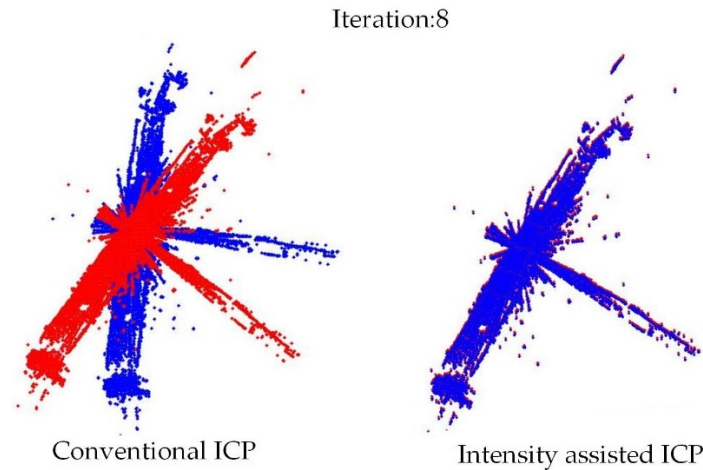


**Figure 7.** Visual comparison of convergence speed for conventional ICP and intensity-assisted ICP. The blue point cloud shows the source frame taken under 0° and red target frame under 30°. On the 8th iteration intensity-assisted ICP has converged but conventional ICP has not.
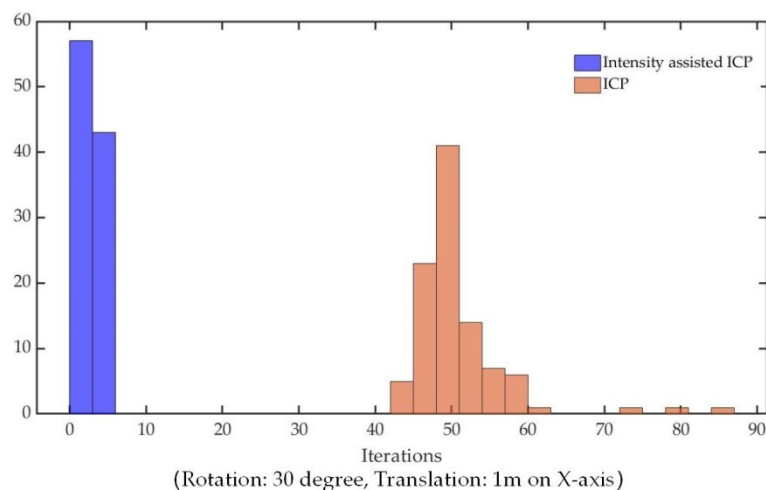


**Figure 8.** The number of iterations required for intensity-assisted ICP and conventional ICP to converge.

Note that the experimental data (containing approximately 20,000 points) are taken randomly about 100 times from the original dataset with the PCL random sample method because the original dataset is too large for ICP operation. After each sampling, the obtained point cloud is used as the source frame, and then the point cloud is transformed as the target frame with rotation: 30°, translation: 1 m on the *X*-axis.

Due to $C_x = C_y = 1$ and $S_x = S_y = 0.2$, the experimental result (rotation: 30°, translation: 1 m on the *Y*-axis) is the same as that (rotation: 30°, translation: 1 m on the *X*-axis). Hence, the experiments in this paper are focused on the situation (rotation: 30°, translation: 1 m on the *X*-axis).

As illustrated by Figure 8, the number of iterations of the proposed method is clearly reduced compared to conventional ICP. That is because of the optimal initial transformation guess and Anderson acceleration.

In addition to the comparison of iterations for convergence, intensity-assisted ICP generates some results with the same quality as given by conventional ICP or even better. This can be seen from

Figure 9, which depicts the error distribution of intensity-assisted ICP and conventional ICP. The error is estimated as the sum of the differences between correspondences in an Euclidean sense, divided by the number of correspondences. Most of runs converge to the same errors, but the number of errors produced from intensity-assisted ICP distributed between 0 and $1 \times 10^{-10}$ are more than that from conventional ICP, which means that intensity-assisted ICP may be more robust than conventional ICP.
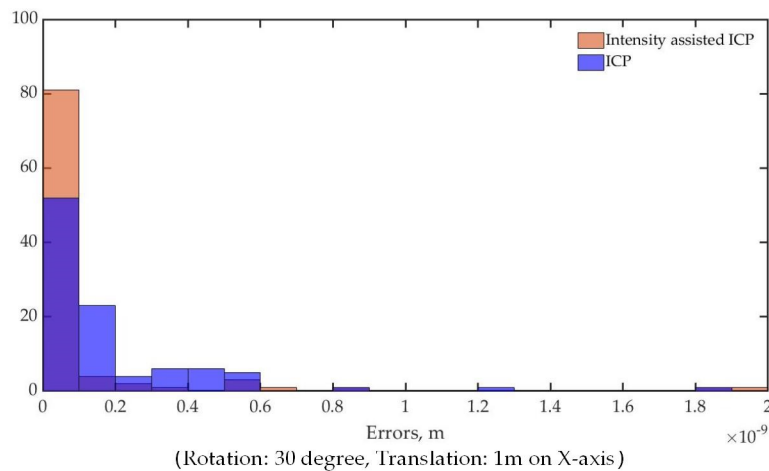


**Figure 9.** The final error convergence of intensity-assisted ICP and conventional ICP.

Due to the process to produce a certain amount of the initial transformation guess, the reduction in number of iterations may not prove the efficiency of the proposed method. The relative change of speed to converge between two algorithms for the same dataset is shown in Figure 10. The speed is calculated from the running time of two algorithms on the same computer. Figure 10 can demonstrate that although the process used to generate the initial transformation guess takes a certain amount of time, the average speed of the intensity-assisted ICP method is still faster than the conventional ICP.
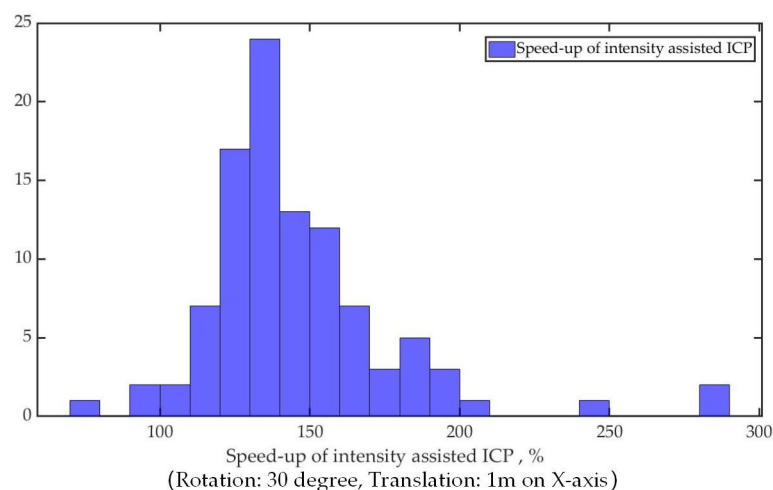


**Figure 10.** The intensity-assisted ICP speed-up relative to the conventional ICP.

To test the acceleration properties of intensity-assisted ICP, 100 random iterations per given degree are introduced and then the relative acceleration is recorded and compared to conventional ICP, where $\varepsilon$ is calculated as 0.001. As can be seen, Figure 11 illustrates the result of the experiments. Relative speed-up of intensity assisted ICP is all higher than that of conventional ICP when rotation increases and finally stay on the slope of approximately 150% when rotation = 45°.
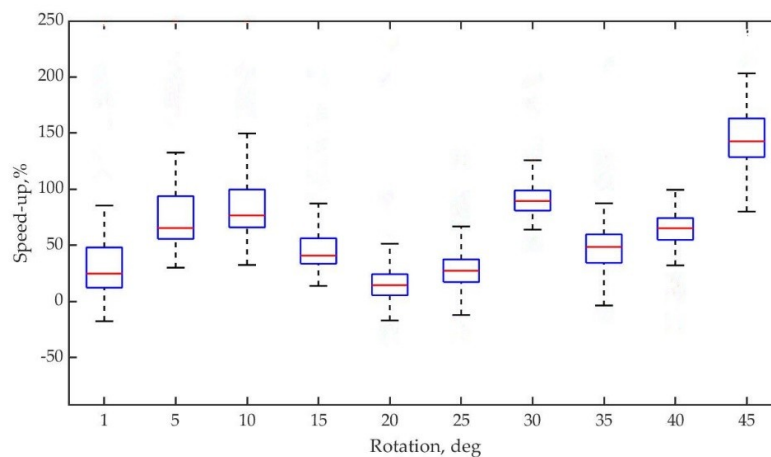
**Figure 11.** Box-plot of relative intensity-assisted ICP speed-up versus random rotation angle for dataset.

The next experiment was conducted for translations. Similarly, 100 random translations per given distance were also introduced. As demonstrated in Figure 12, it looks quite similar to the experiments with random rotations. Relative speed-up of intensity-assisted ICP is all higher than that of conventional ICP when distance increases and can stay on the slope of approximately 50% when translation = 0.95 m.
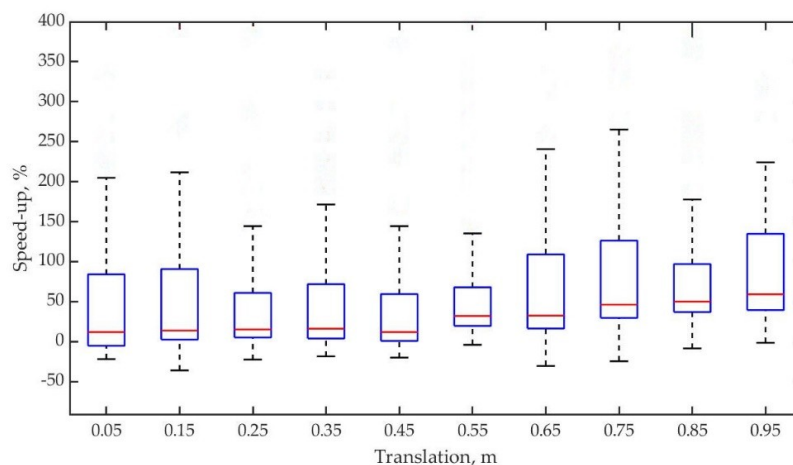


**Figure 12.** Box-plot of relative intensity assisted ICP speed-up versus random translation distance for dataset.

Note that the large fluctuation of relative speed-up originates from different point clouds per sampling operation.

Finally the intensity-assisted ICP is compared with conventional ICP, AA-ICP and implicit moving least squares-ICP (IMLS-ICP) [30] with datasets Market (marketsquarefeldkirch4-reduced), Stgallen (stgallencathedral6-reduced), Station (untermaederbrunnen1) extracted from SEMANTIC3D.NET using the downsampling method with the sampled grid size 0.1 m different from the rand sampling method above. The three original datasets can be seen in Figure 13.

As shown in Table 1, the intensity-assisted ICP performs well in the three datasets even better than experiments above. The reason is that the sampling method is downsampling in this step but in the experiments above it is rand sampling. The intensity-assisted ICP can achieve faster speed and better accuracy compared to conventional ICP and AA-ICP. The IMLS-ICP algorithm [30] is a state-of-the-art ICP algorithm introduced in 2018 and ranked third in the KITTI dataset. The remarkable feature of the IMLS-ICP algorithm is its high accuracy. The intensity-assisted ICP is also faster to converge

than IMLS-ICP but with low accuracy. This may be a disadvantage of intensity-assisted ICP, but the intensity-assisted ICP can still maintain excellent performance compared to other algorithms.
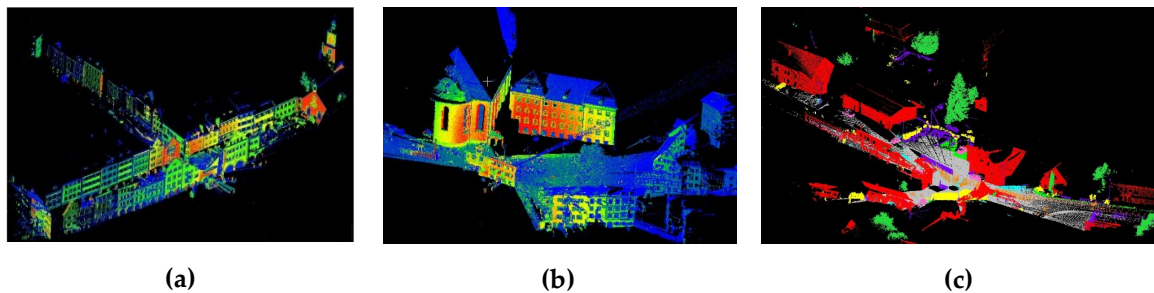


**(a)**　　　　　　　　　　　**(b)**　　　　　　　　　　　**(c)**

**Figure 13.** (**a**) The Market datasets (marketsquarefeldkirch4-reduced). (**b**) The Stgallen datasets (stgallencathedral6-reduced). (**c**) The Station datasets (untermaederbrunnen1).

**Table 1.** Comparison of the speed and accuracy of four ICP algorithms.

| Dataset | Parameters | Algorithms | | | |
|---------|-----------|------|--------|----------|---------|
| | | ICP | AA-ICP | IMLS-ICP | Our ICP |
| Market | Experiment1 (1 m, 0 m, 30°) | | | | |
| | Errors (m) | $5.808 \times 10^{-3}$ | $3.644 \times 10^{-3}$ | $1.317 \times 10^{-4}$ | $1.556 \times 10^{-3}$ |
| | Time (s) | 125.936 | 76.455 | 45.469 | 12.674 |
| | Experiment2 (0.5 m, 0 m, 15°) | | | | |
| | Errors (m) | $5.806 \times 10^{-3}$ | $3.261 \times 10^{-3}$ | $3.746 \times 10^{-5}$ | $2.185 \times 10^{-3}$ |
| | Time (s) | 56.454 | 23.113 | 42.924 | 13.901 |
| | Experiment3 (0.1 m, 0 m, 10°) | | | | |
| | Errors (m) | $1.755 \times 10^{-3}$ | $1.757 \times 10^{-3}$ | $3.293 \times 10^{-4}$ | $1.659 \times 10^{-9}$ |
| | Time (s) | 54.458 | 37.008 | 48.154 | 22.093 |
| Stgallen | Experiment1 (1 m, 0 m, 30°) | | | | |
| | Errors (m) | $8.662 \times 10^{-3}$ | $5.891 \times 10^{-3}$ | $1.523 \times 10^{-4}$ | $3.265 \times 10^{-3}$ |
| | Time (s) | 239.642 | 168.634 | 138.276 | 21.272 |
| | Experiment2 (0.5 m, 0 m, 15°) | | | | |
| | Errors (m) | $6.039 \times 10^{-3}$ | $1.971 \times 10^{-3}$ | $6.603 \times 10^{-5}$ | $1.853 \times 10^{-3}$ |
| | Time (s) | 158.853 | 70.818 | 121.159 | 23.416 |
| | Experiment3 (0.1 m, 0 m, 10°) | | | | |
| | Errors (m) | $4.159 \times 10^{-3}$ | $9.012 \times 10^{-3}$ | $4.203 \times 10^{-5}$ | $1.942 \times 10^{-3}$ |
| | Time (s) | 156.046 | 59.321 | 114.177 | 20.718 |
| Station | Experiment1 (1 m, 0 m, 30°) | | | | |
| | Errors (m) | $3.096 \times 10^{-2}$ | $1.527 \times 10^{-2}$ | $2.417 \times 10^{-5}$ | $2.357 \times 10^{-3}$ |
| | Time (s) | 57.688 | 47.728 | 56.993 | 12.524 |
| | Experiment2 (0.5 m, 0 m, 15°) | | | | |
| | Errors (m) | $1.368 \times 10^{-2}$ | $1.079 \times 10^{-2}$ | $2.387 \times 10^{-4}$ | $9.748 \times 10^{-4}$ |
| | Time (s) | 51.574 | 37.425 | 64.251 | 12.576 |
| | Experiment3 (0.1 m, 0 m, 10°) | | | | |
| | Errors (m) | $4.911 \times 10^{-3}$ | $8.655 \times 10^{-3}$ | $9.817 \times 10^{-5}$ | $1.049 \times 10^{-4}$ |
| | Time (s) | 54.861 | 41.817 | 42.546 | 12.876 |

## 5. Conclusions

As discussed above, the intensity-assisted ICP is proposed and analyzed. A novel modification of the conventional ICP based on the initial transformation guess and Anderson acceleration indicates the proposed method can perform well in this paper. On the initial transformation guess stage, this new method improves the conventional ICP significantly. A target function is proposed to determine the best initial transformation guess based on the statistic of their spatial distances and intensity residuals with a little runtime cost. It reduces substantially the number of iterations required for achieving desired matching quality compared to the conventional ICP. Besides that, the Anderson acceleration is also used to increase the speed of iteration which has been considered as having better performance compared to the conventional Picard iteration procedure. Moreover, the SEMANTIC3D.NET benchmark dataset is used to evaluate the proposed method. The experimental results show that the new method presented in this paper achieves better speed and quality for convergence compared to the conventional ICP.

## References

1. Cadena, C.; Carlone, L.; Carrillo, H.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; Leonard, J. Past, present, and future of simultaneous localization and mapping: toward the robust-perception age. *IEEE Trans. Robot.* **2016**, *32*, 1309–1332. [CrossRef]
2. Besl, P.J.; McKay, N.D. A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* **1992**, *14*, 239–256. [CrossRef]
3. Tang, J.; Chen, Y.; Jaakkola, A.; Liu, J.; Hyyppä, J.; Hyyppä, H. NAVIS—An UGV indoor positioning system using laser scan matching for large-area real-time applications. *Sensors* **2014**, *14*, 11805–11824. [CrossRef] [PubMed]
4. Im, J.H.; Im, S.H.; Jee, G.I. Vertical Corner Feature Based Precise Vehicle Localization Using 3D LIDAR in Urban Area. *Sensors* **2016**, *16*, 1268. [CrossRef] [PubMed]
5. Almeida, J.; Santos, V.M. Real time egomotion of a nonholonomic vehicle using LIDAR measurements. *J. Field Robot.* **2012**, *30*, 129–141. [CrossRef]
6. Kuramachi, R.; Ohsato, A.; Sasaki, Y.; Mizoguchi, H. G-ICP SLAM: An odometry-free 3D mapping system with robust 6DoF pose estimation. In Proceedings of the IEEE International Conference on Robotics & Biomimetics, Zhuhai, China, 6–9 December 2015.
7. Yang, Y.; Yang, G.; Zheng, T.; Tian, Y.; Li, L. Feature extraction method based on 2.5-dimensions lidar platform for indoor mobile robots localization. In Proceedings of the 2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), Ningbo, China, 19–21 November 2017; pp. 736–741.
8. Yang, Y.; Yang, G.; Tian, Y.; Zheng, T.; Li, L.; Wang, Z. A robust and accurate SLAM algorithm for omni-directional mobile robots based on a novel 2.5D lidar device. In Proceedings of the 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), Wuhan, China, 31 May–2 June 2018; pp. 2123–2127.
9. Witt, J.; Weltin, U. Robust Stereo Visual Odometry Using Iterative Closest Multiple Lines. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots & Systems, Tokyo, Japan, 3–7 November 2013.
10. Zhao, S.; Fang, Z. Direct Depth SLAM: Sparse Geometric Feature Enhanced Direct Depth SLAM System for Low-Texture Environments. *Sensors* **2018**, *18*, 3339. [CrossRef] [PubMed]

11. Yang, J.; Li, H.; Jia, Y. Go-ICP: Solving 3D Registration Efficiently and Globally Optimally. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 3–6 December 2013.

12. Censi, A. An ICP variant using a point-to-line metric. In Proceedings of the IEEE International Conference on Robotics & Automation, Pasadena, CA, USA, 19–23 May 2008.

13. Burlacu, A.; Cohal, A.; Caraiman, S.; Condurache, D. Iterative closest point problem: A tensorial approach to finding the initial guess. In Proceedings of the International Conference on System Theory, Sinaia, Romania, 13–15 October 2016.

14. Attia, M.; Slama, Y.; Attia, M.; Slama, Y.; Attia, M.; Slama, Y. Efficient Initial Guess Determination Based on 3D Point Cloud Projection for ICP Algorithms. In Proceedings of the International Conference on High Performance Computing & Simulation, Genoa, Italy, 17–21 July 2017.

15. Men, H.; Gebre, B.; Pochiraju, K. Color point cloud registration with 4D ICP algorithm. In Proceedings of the IEEE International Conference on Robotics & Automation, Shanghai, China, 9–13 May 2011.

16. Serafin, J.; Grisetti, G. NICP: Dense Normal Based Point Cloud Registration. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots & Systems, Hamburg, Germany, 28 September–2 October 2015.

17. Greenspan, M.; Yurick, M. Approximate k-d tree search for efficient ICP. In Proceedings of the International Conference on 3-d Digital Imaging & Modeling, Banff, AB, Canada, 6–10 October 2003.

18. Granger, S.; Pennec, X. Multi-scale EM-ICP: A Fast and Robust Approach for Surface Registration. In Proceedings of the European Conference on Computer Vision, Copenhagen, Denmark, 28–31 May 2002.

19. Rusinkiewicz, S.; Levoy, M. Efficient Variants of the ICP Algorithm. In Proceedings of the International Conference on 3-d Digital Imaging & Modeling, Quebec, QC, Canada, 28 May–1 June 2001.

20. Savva, A.D.; Economopoulos, T.L.; Matsopoulos, G.K. Geometry-based vs. intensity-based medical image registration: A comparative study on 3D CT data. *Comput. Biol. Med.* **2016**, *69*, 120–133. [CrossRef] [PubMed]

21. Li, S.; Lee, D. Fast Visual Odometry Using Intensity-Assisted Iterative Closest Point. *IEEE Robot. Autom. Lett.* **2016**, *1*, 992–999. [CrossRef]

22. Smith, E.R.; Kinga, B.J.; Stewarta, C.V.; Radkea, R.J. Registration of combined range–intensity scans: Initialization through verification. *Comput. Vis. Image Underst.* **2008**, *110*, 226–244. [CrossRef]

23. Kim, H.; Liu, B.; Chi, Y.G.; Lee, S.; Myung, H. Robust Vehicle Localization using Entropy-weighted Particle Filter-based Data Fusion of Vertical and Road Intensity Information for a Large Scale Urban Area. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1518–1524. [CrossRef]

24. Pavlov, A.L.; Ovchinnikov, G.V.; Derbyshev, D.Y.; Tsetserukou, D.; Oseledets, I.V. AA-ICP: Iterative Closest Point with Anderson Acceleration. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018.

25. Walker, H.F.; Ni, P. Anderson Acceleration for Fixed-Point Iterations. *Math. Sci. Fac. Publ.* **2007**, *49*, 1715–1735.

26. Potra, F.A.; Engler, H. A characterization of the behavior of the Anderson acceleration on linear problems. *Linear Algebra Its Appl.* **2013**, *438*, 1002–1011. [CrossRef]

27. Kerl, C.; Sturm, J.; Cremers, D. Robust odometry estimation for RGB-D cameras. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 6–10 May 2013.

28. Lange, K.; Little, R.A.; Taylor, J.G. Robust Statistical Modeling Using the t Distribution. *Publ. Am. Stat. Assoc.* **1989**, *84*, 881–896. [CrossRef]

29. Hackel, T.; Savinov, N.; Ladicky, L.; Wegner, J.D.; Schindler, K.; Pollefeys, M. Semantic3D.net: A new Large-scale Point Cloud Classification Benchmark. *arXiv* **2017**, arXiv:1704.03847. [CrossRef]

30. Deschaud, J.E. IMLS-SLAM: scan-to-model matching based on 3D data. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, Australia, 21–25 May 2018; pp. 2480–2485.