



# Article Comparative Analysis of 3D LiDAR Scan-Matching Methods for State Estimation of Autonomous Surface Vessel

Haichao Wang , Yong Yin and Qianfeng Jing \*

Key Laboratory of Marine Simulation and Control, Department of Navigation, Dalian Maritime University, Dalian 116026, China; wanghaichao@dlmu.edu.cn (H.W.); bushyin@dlmu.edu.cn (Y.Y.) \* Correspondence: jqf\_dlmu@dlmu.edu.cn

Abstract: Accurate positioning and state estimation of surface vessels are prerequisites to achieving autonomous navigation. Recently, the rapid development of 3D LiDARs has promoted the autonomy of both land and aerial vehicles, which has aroused the interest of researchers in the maritime community accordingly. In this paper, the state estimation schemes based on 3D LiDAR scan matching are explored in depth. Firstly, the iterative closest point (ICP) and normal distribution transformation (NDT) algorithms and their variants are introduced in detail. Besides, ten representative registration algorithms are selected from the variants for comparative analysis. Two types of experiments are designed by utilizing the field test data of an ASV equipped with a 3D LiDAR. Both the accuracy and real-time performance of the selected algorithms are systemically analyzed based on the experimental results. It follows that ICP and Levenberg–Marquardt iterative closest point (LMICP) methods perform well on single-frame experiments, while the voxelized generalized iterative closest point (FastVGICP) and multi-threaded optimization generalized iterative closest point (FastGICP) methods have the best performance on continuous-frame experiments. However, all methods have lower accuracy during fast turning. Consequently, the limitations of current methods are discussed in detail, which provides insights for future exploration of accurate state estimation based on 3D LiDAR for ASVs.

Keywords: 3D LiDAR; ASV; scan matching; registration; state estimation



Citation: Wang, H.; Yin, Y.; Jing, Q. Comparative Analysis of 3D LiDAR Scan-Matching Methods for State Estimation of Autonomous Surface Vessel. J. Mar. Sci. Eng. 2023, 11, 840. https://doi.org/10.3390/jmse11040840

Academic Editors: Alessandro Ridolfi, Bruno Miguel Ferreira and Nuno A. Cruz

Received: 13 March 2023 Revised: 7 April 2023 Accepted: 13 April 2023 Published: 15 April 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). 1. Introduction

Autonomous surface vehicles (ASVs) have the characteristics of high autonomy and intelligence [1] and can perform a variety of high-risk tasks in harsh or inaccessible marine environments [2]. Its low weight and compact size make it highly mobile [3]. Since there is no crew onboard, personnel safety is higher [4]. Besides, they can also be deployed in numerous applications involving many fields, such as the civil, industrial, and military fields [5], such as environmental monitoring [6], coastal surveillance [4], resource exploration [7], reconnaissance [8], and rescue [9]. In general, ASVs are extremely suitable for eliminating personal hazards and improving safety in dangerous and rough marine missions. In addition, it is expected that the market for ASVs will proliferate shortly.

State estimation is one of the most studied issues in the field of ASV [10]. It is a prerequisite for ships to achieve genuine autonomy. At present, most ships mainly rely on the global navigation satellite system (GNSS) and inertial measurement unit (IMU) to obtain their pose. When the ship is sailing in the harbor, GNSS performance is seriously affected by the multipath (MP) effect and the non-line-of-sight (NLOS) reception, and the positioning error may increase to several meters [11] or even tens of meters [12]. IMU has a cumulative error, and the reliability decreases with time [13]. Therefore, an alternative state estimation system is needed to estimate the ship's motion by using the surrounding environment. When the GNSS and IMU information is inaccurate or interrupted, it provides accurate position and motion estimation for the ship to avoid collisions and other accidents.

To solve the above problems, ASV usually relies on the data collected by its sensors to apply the Simultaneous Localization and Mapping (SLAM) method for positioning. The sensors used in ASV mainly include GNSS, IMU, gyro compass, magnetic compass, camera, radar, LiDAR, geomagnetic direction sensor (GDS), and fiber-optic gyro (FOG). GNSS, IMU, camera, radar, and LiDAR can be used for ship positioning. Gyro compass, magnetic compass, GDS, and FOG are mainly used for azimuth measurement. In addition, GNSS and IMU are the basic sensors to ensure the safe operation of USV, which can be called safety sensors. Camera [14], radar [15], and LiDAR [16] can use the scan matching between consecutive frames for ASV state estimation. However, the camera is easily affected by illumination and weather conditions, resulting in large registration errors and inaccurate state estimation [17]. The low resolution of radar output also affects the accuracy of registration and state estimation [18]. Compared with the camera and radar, LiDAR can provide reliable and accurate distance measurement and is more robust to the environment with poor lighting conditions or less optical texture [19]. State estimation accuracy is usually higher. It has been a research hotspot in recent years. This paper mainly describes the autonomous positioning scheme based on 3D LiDAR.

Scan matching is the core module of the LiDAR SLAM [20]. It uses the relationship between adjacent frames to estimate the motion pose of the current frame. Depending on this process, the current robot pose will be incrementally updated. Scan matching is commonly used in positioning and six-degree-of-freedom (6DOF) state estimation of unmanned aerial vehicles [21], unmanned ground vehicles [22], and ASV [23]. For some applications that require precise positioning, scan matching can further refine the positioning accuracy, especially for scenarios such as berthing that require high position and rotation accuracy.

Iterative closest point (ICP) [24] and normal distribution transformation (NDT) [25] are classic scan-matching algorithms. The ICP algorithm aligns two frames of point clouds by distance judgment, and the NDT algorithm uses a Gaussian distribution to model point clouds. Both ICP and NDT are regarded as local scan-matching methods that begin with an initial transformation and then iterate until a local minimum is reached [26]. In addition, there are some feature-based matching methods. The features used mainly include line-plane features [27], viewpoint feature histograms [28], fast point feature histograms [29], 3D neighborhood point feature histograms [30], density characteristics [31], etc. The most representative method is the LOAM [27] scheme, which uses corner and plane point features.

Experts and scholars have done a lot of work in scanning matching research [32]. Some work is for specific scenarios [33]. There are also studies on the comparative analysis of different algorithms, including traditional methods [34] and deep learning methods [35]. However, these studies mainly focus on the situation on land and do not explore the application of these algorithms on ASV. Compared with the land field, the situation at sea is more complicated, including wind, waves, currents, sea fog, etc. [36]. In addition, the motion pose of the ship at sea is also more nonlinear. Compared with vehicles constrained by ground and wheels, ASV is completely six-degree-of-freedom motion on the water surface, which poses a challenge to 3D LiDAR scanning matching [37]. In view of the above situation, this paper develops a ship state estimation framework based on a scanning registration algorithm. By comparing and evaluating 10 algorithms, a scanning registration algorithms in marine applications, which can provide a reference for the following research. The main contributions of this article are summarized as follows:

- (1) This paper discusses ICP and NDT algorithms and their variants. A single-frame point cloud registration scheme based on PCL is built. Considering the accuracy and registration time, the appropriate parameters are selected for the algorithm, and 10 common registration algorithms are tested and analyzed on ASV data.
- (2) Secondly, a continuous-frame point cloud registration module based on ROS is built. The results are compared with RTK and IMU values, and the performance of the algorithm is analyzed from multiple perspectives. In addition, the problem of large local errors of roll and pitch obtained by the continuous-frame registration module is studied, and the causes of errors are analyzed from many aspects.

(3) Finally, the registration algorithm is verified by an experiment with real ASV, and the single-frame and continuous-frame scanning registration schemes suitable for marine environments are obtained.

# 2. ICP and NDT

2.1. ICP

The basic model of the ICP algorithm is: given source point cloud  $P = \{p_1, p_2, p_3, \dots, p_m\}$ and target point cloud  $Q = \{q_1, q_2, q_3, \dots, q_n\}$ , P,  $Q \subset R^3$ , m,n are the number of source point cloud and target point cloud, respectively. Let E(T) be the error between the source point cloud P and the target point cloud Q under the transformation matrix T. Then E(T)can be expressed by the following formula:

$$E(T) = E(R, t) = \sum_{i=1}^{m} \|(Rp_i + t) - q_i\|^2 \quad T = \begin{pmatrix} R & t \\ 0^T & 1 \end{pmatrix}$$
(1)

where R is the rotation matrix, t is the translation matrix,  $p_i$  is the point in the source point cloud, and  $q_i$  is the nearest point in the target point cloud.

Solve *R*, *t* by minimizing the above error function.

$$\boldsymbol{R}, \boldsymbol{t} \leftarrow \operatorname*{argmin}_{\boldsymbol{R}, \boldsymbol{t}} \sum_{i=1}^{m} \| (\boldsymbol{R} \boldsymbol{p}_i + \boldsymbol{t}) - \boldsymbol{q}_i \|^2$$
(2)

The specific process is as following:

```
1: Input: Source Point Cloud P = \{p_1, p_2, p_3, \dots, p_m\}
        Target Point Cloud Q = \{q_1, q_2, q_3, \dots, q_n\}
2:
3:
        Initial Transformation T_0
4: Output: Correct transformation matrix T matching P and Q
5:
    T \leftarrow T_0
     While the result does not converge Do
6:
7:
            for i \leftarrow 1 to m
               q_i \leftarrow FindClosestPointInQ(T \cdot p_i)
8:
9:
               if \|T \cdot p_i - q_i\| \le d_{max} then
                   w_i \leftarrow 1;
10:
11:
                else
                   w_i \leftarrow 0;
12:
                end if
13:
             end for
14:
              T \leftarrow \operatorname{argmin}_{T} \left\{ \sum_{i} w_{i} \| T p_{i} - q_{i} \|^{2} \right\}
15:
16: end while
```

#### 2.2. NDT

The basic model of the NDT algorithm is: Given source point cloud  $P = \{p_1, p_2, p_3, ..., p_m\}$  and target point cloud  $Q = \{q_1, q_2, q_3, ..., q_n\}$ , P,  $Q \subset R^3$ . Firstly, the NDT algorithm divides the space covered by the target point cloud into voxels. Then, based on the distribution of the midpoints of each voxel, the probability density function (PDF) is calculated for each voxel. Suppose that the position of the point on the target point cloud is obtained by random processing of the D-dimensional normal distribution. The probability function of the point *x* in each voxel is:

$$F(x) = \frac{1}{(2\pi)^{\frac{D}{2}}\sqrt{|\boldsymbol{\Sigma}|}} \exp\left(-\frac{(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}{2}\right)$$
(3)

where *D* represents the dimension,  $\mu$  and  $\Sigma$  represent the mean vector and covariance matrix of all points in the voxel where *x* is located, respectively,  $\mu$  and  $\Sigma$  are calculated by the following formulas:

$$\boldsymbol{\mu} = \frac{1}{k} \sum_{i=1}^{k} \boldsymbol{q}_i \tag{4}$$

$$\boldsymbol{\Sigma} = \frac{1}{k-1} \sum_{i=1}^{k} (\boldsymbol{q}_i - \boldsymbol{\mu}) (\boldsymbol{q}_i - \boldsymbol{\mu})^T$$
(5)

 $q_{i=1,\dots,k}$  is the position coordinates of all points contained in the voxel.

Finally, the pose transformation T is obtained, so that the overall likelihood of the transformed source point cloud in its corresponding probability density function reaches the maximum, that is

$$\boldsymbol{R}, \boldsymbol{t} \leftarrow \operatorname*{argmax}_{\boldsymbol{R}, \boldsymbol{t}} \prod_{i=1}^{m} F(\boldsymbol{R}\boldsymbol{p}_{i} + \boldsymbol{t})$$
(6)

Using the Newton iteration method, find the optimal transformation parameters to complete the point cloud registration.

```
1: Input: Source Point Cloud P = \{p_1, p_2, p_3, \dots, p_m\}
2:
          Target Point Cloud Q = \{q_1, q_2, q_3, \dots, q_n\}
3:
          Initial Transformation T_0
4: Output: Correct transformation matrix T matching P and Q
5: T \leftarrow T_0
      Target point cloud space voxel segmentation V = \{v_1, v_2, \cdots, v_l\}
6:
7:
       for i \leftarrow 1
8:
             Find voxel v_k containing point q_i
9٠
             Store q_i in v_k
10: end for
11:
         for j \leftarrow 1 to l
               \mathbf{Q}' = \{\mathbf{q}'_1, \mathbf{q}'_2, \cdots, \mathbf{q}'_{\mathbf{n}'}\} \leftarrow v_j
12:
13:
               if n' \leq 5 then
                                           // Delete grid cells with too few elements
14:
                   delete v_i
15:
               else
                   \mu_i \leftarrow \frac{1}{n'} \sum_{k=1}^{n'} q'_k
16:
                   \boldsymbol{\Sigma}_{j} \leftarrow rac{1}{n'-1} \sum_{k=1}^{n'} \left( \boldsymbol{q}_{k}' - \boldsymbol{\mu}_{j} 
ight) \left( \boldsymbol{q}_{k}' - \boldsymbol{\mu}_{j} 
ight)^{T}
17:
18:
               end if
19:
       end for
20:
        T \leftarrow \operatorname{argmax}_{T} \prod_{i=1}^{m} F(Tp_i)
```

#### 3. Point Cloud Registration Method

Point cloud registration accuracy is positively correlated with state estimation accuracy. The higher the registration accuracy, the more accurate the state estimation. To improve registration accuracy, new registration algorithms and their variants are proposed. This chapter discusses variants based on ICP and NDT, respectively.

## 3.1. ICP-Based Variants

The ICP algorithm has some shortcomings, such as sensitivity to the initial value [38], high computational overhead [39], being easily trapped in a local optimal solution [40], and a lack of utilization of point cloud structure information [41]. In response to these problems, experts and scholars have exploited many improvement methods. Censi [42] proposed the PLICP algorithm, using the point-line distance metric to replace the point-to-point distance metric in the ICP algorithm. Compared with P2P-ICP, PLICP has higher accuracy and requires fewer iterations, but PLICP is more susceptible to initial values. Chen and Medioni [43] proposed the P2PL-ICP algorithm, which improved the selection method

of the corresponding points in the ICP algorithm (P2P-ICP), and used the distance from the point to the surface instead of the distance from the point to the point to improve the convergence speed. Segal et al. [44] presented the Generalized-ICP (GICP) algorithm, which combines the P2P-ICP algorithm and the P2PL-ICP algorithm into a probabilistic framework based on which point cloud matching is carried out. The covariance matrix plays a role analogous to the weight to eliminate some bad corresponding points in the solution process. The GICP algorithm can flexibly model different models, the P2P-ICP algorithm is equivalent to a unique form of the GICP algorithm. GICP algorithm can effectively reduce the impact of mismatch, and it constitutes one of the most effective and robust algorithms among many improved ICP algorithms. Based on GICP, Koide et al. [45] extended the GICP method by voxelization to improve the registration speed while maintaining its accuracy. In addition, the multi-threaded CPU GICP algorithm is implemented. Sharp et al. [46] proposed the ICP algorithm using invariant features (ICPIF) by combining Euclidean invariant features with the ICP algorithm. The algorithm uses the distance function of the weighted linear combination of position distance and feature distance to obtain the corresponding relationship. The experimental results demonstrate that the intersection is more accurate than the corresponding relationship using only position distance. Minguez et al. [47] used a new distance scale function to consider both translation and rotation. Compared with the ICP algorithm, the MbICP algorithm has improved robustness, accuracy, convergence, and computational load. Serafin and Grisetti [48] used surface features such as the normal vector and curvature to filter out error points during matching. In the iterative solution process, the error function not only includes the projection distance of the normal vector between point clouds (same P2PL-ICP) but also includes the normal vector direction error. Experimental analysis shows that the NICP algorithm has better accuracy and robustness.

Granger and Pennec [49] developed the EM-ICP algorithm and applied the EM algorithm to the ICP algorithm to avoid the initial registration step. Aiming at the point cloud distortion caused by motion (especially high-speed motion). Hong et al. [50] introduced speed updates based on the ICP algorithm to obtain more accurate pose estimation. According to the distortion effect caused by sensor motion during point cloud registration, Alismail et al. [51] designed a Continuous-ICP (CICP) algorithm, which uses the estimated continuous 6DOF pose of the sensor to correct the distortion and improves the accuracy and robustness of the algorithm.

The ICP algorithm is easy to converge to the local minimum. To solve this problem, Yang et al. [52] combined the ICP algorithm with the branch-and-bound method to ensure a global optimal solution. Although this method solves the local minimum problem, it is always sensitive to initialization and has a large computational cost. As to the challenge of accurate registration of outdoor large-scale multi-robot environment data fusion, Han et al. [53] proposed an enhanced ICP method that combines a hierarchical search scheme with an octree-based ICP algorithm to achieve coarse-to-fine registration of large-scale multi-resolution data. An early warning mechanism is utilized to sense local minimum problems. A heuristic escape scheme based on sampling potential transformation vectors to avoid local extremum. The superior performance of the exploited algorithm is checked by experiments.

Kdtree can be used in the process of searching for the nearest point to optimize and reduce the search time [54,55]. Fitzgibbon [56] used the general nonlinear optimization (Levenberg–Marquardt algorithm) to directly minimize the registration error. Pavlov et al. [57] designed an ICP algorithm using Anderson acceleration (AA-ICP), which accelerates ICP by modifying the iterative process and improves the convergence speed and effect of registration. Zhang et al. [58] proposed a new robust registration method with fast convergence. This method accelerates its convergence by using the Anderson acceleration method. In addition, the Welsch function is introduced to improve the robustness of ICP.

For the problem that the ICP algorithm is sensitive to outliers and missing data, Bouaziz et al. [59] improved the ICP algorithm and used the sparsity induction norms to optimize the registration, which improved the robustness of ICP to outliers and incomplete point cloud data. Agamennoni et al. [60] elaborated an improved standard ICP data association strategy, which associates each point in the source point cloud with a set of points in the target point cloud. Experimental analysis shows that the improved method can effectively deal with the alignment problem between dense point clouds and sparse point clouds. TrimmedICP [61] introduced a solution for partially overlapped point cloud registration, using least-trimmed squares to solve the ICP problem, which improves the robustness to noise and allows two-point cloud sets to have poor initial values of transformation parameters. Rusinkiewicz [62] proposed a symmetric objective function for ICP, which realizes the simplicity and computational efficiency of point-to-plane optimization and improves the convergence speed and wider convergence range. Du et al. [63] and Wu et al. [64] introduced the maximum correntropy criterion (MCC) as a robust metric for point-set rigid registration. The ICP algorithm based on MCC (MCC-ICP) can deal with noise and outliers well and shows its superiority in accuracy and robustness. A recent work [65] introduced soft constraints as the cost of ICP optimization, allowing elastic deformation of the scan during registration to improve accuracy and robustness to high-frequency motion from discontinuities.

## 3.2. NDT-Based Variants

NDT was originally presented by Biber and Strasser for 2D scan matching [25]. Magnusson et al. [66] extended 2D NDT to 3D. Magnusson et al. [67] conducted a comprehensive comparison of the 3D scan-matching algorithms of ICP and NDT. The experimental results demonstrate that NDT performs faster than ICP and converges from a wider range of initial pose estimations. However, the NDT convergent pose is unpredictable, and the pose generated in the case of failure is worse than the ICP. The author proposes an NDT algorithm with trilinear interpolation to solve this problem. This algorithm has a higher success rate than the classical NDT algorithm, but it takes a longer time. To speed up NDT registration, Koide et al. [68] encapsulated SSE-friendly and multi-threaded NDT, which can run up to 10 times faster than its original version in PCL. In addition, three methods for neighbor voxel search are provided, including KDTREE, DIRECT1, and DIRECT7.

Zaganidis et al. [69] added semantic labels to point clouds by calculating the smoothness of point clouds. Point clouds without semantic labels are discarded and do not take part in registration. Point clouds with distinct semantic labels are separately registered by NDT. Experimental analysis has shown that the algorithm improves the speed and robustness of registration. Jun et al. [70] developed an NDT registration algorithm with variable voxel size that considers the local density in the process of dividing a large voxel into several small voxels and gathers the sparse point cloud as a large voxel. The point cloud is densely dispersed into multiple small voxels to solve the problem that most sparse points cannot be used due to the difference in the number of point clouds caused by the fixed voxel size and enhance the accuracy of registration. Hong and Lee [71] converted the reference point cloud into a disclike distribution suitable for the point cloud structure, addressing the effect of the thick distributions whose mass centers are hanging in the air on NDT registration.

Ulas and Temeltas [72] introduced a multi-layer-based transformation algorithm for the normal distribution. The algorithm automatically calculates the cell size by looking at the boundary of the point cloud, and then the point cloud is subdivided into 8n cells, where n represents the level of the layer. In addition, the method uses the Mahalanobis distance function as the score function to replace the original Gaussian probability function and optimization using the Newton and Levenberg–Marquardt methods. Compared with the NDT algorithm, this method has longer-distance measurement capability and a faster convergence speed. Later, Ulas and Temeltas [73] added a feature extraction module to the above algorithm, namely MLNDT based on feature extraction, which further improved the efficiency and robustness of the registration algorithm. Hong and Lee [74] presented the KLNDT algorithm to solve the problems of the fixed number of layers and the limited number of optimization iterations in MLNDT. The algorithm searches the key layer and registers in the key layer until the termination criterion is satisfied. Experiments show that KLNDT has a higher success rate and accuracy than MLNDT. Hong and Lee [75] presented PNDT, defined the probability of point samples, and calculated the mean and covariance according to the probability. Compared with the classical NDT, PNDT point cloud registration accuracy is higher. Zaganidis et al. [76] used semantic information extracted from point clouds to enrich NDT features and combined NDT histogram descriptors for loop closure detection. This method has higher accuracy than non-semantic NDT methods. Lee et al. [77] proposed a weighted normal distribution transform (weightedNDT) method, which reflects the static probability of each point as a weight. Compared with his dynamic target removal method, the weightedNDT algorithm has higher classification accuracy and lower scanning matching error. Deng et al. [78] introduced an optimized normal distribution transform algorithm (SEO-NDT) and its FPGA implementation. Compared with the state-of-the-art embedded CPU and GPU implementations, the scheme performs faster and can run in real time when processing larger point clouds. Aiming at the problem of poor real-time performance and pose drift error of the NDT algorithm in large scenes, Zhong et al. [79] proposed a factor graph optimization algorithm (FGO-NDT). The experimental results show that this method can eliminate the attitude estimation error caused by drift, improve the local structure, and reduce the root mean square error.

## 4. Experiments

In order to evaluate the performance of the above algorithms, this chapter tests the common registration algorithms, including ICP, LMICP, GICP, multi-threaded optimization GICP (FastGICP), voxelized GICP (FastVGICP), point-to-plane ICP (PTPLICP). Multithreading accelerated PTPLICP (PTPLOMPICP), NDT, multithreading optimized NDT and the DIRECT1 method for neighbor voxel search (NDTOMP1), and multithreading optimized NDT and the DIRECT7 method for neighbor voxel search (NDTOMP7).

#### 4.1. Experimental Environment and Equipment Configuration

The experimental site is "Lingshui Port" in Dalian, China. The experimental ASV is "Zhi Long No 1", which is 1.75 m long and 0.5 m wide, with a double propeller and double rudder. The ASV is equipped with a rfans-16 mechanical LiDAR, RTK, high-precision GPS/IMU, and a stereo camera. The experiment's weather is cloudy, with a north wind level of 3~4 and a wave level of 1. For port information and other equipment-specific configurations, please see the paper [80]. All methods were implemented on an Intel Core i5-4570 with 16 GB of RAM and an NVIDIA GeForce GTX1650.

## 4.2. Experimental Design

The experiment selects the navigation data of the ASV in the port. The algorithm performance test experiment is divided into two groups, namely the single-frame experiment and the continuous-frame experiment. In the single-frame experiment, 10 pairs of data are selected from the data of the whole navigation stage, and each pair of data is two adjacent frames. The average registration time and root mean square error (*RMSE*) of ten algorithms on 10 pairs of data are calculated. The purpose of the single-frame algorithm is to test the registration accuracy of each algorithm and select appropriate parameters for each algorithm. The continuous-frame experiment is designed to continuously register each frame data on the entire navigation data set and obtain the final result. In this experiment, the performance of each registration algorithm is evaluated by calculating the absolute trajectory error (ATE) and relative pose error (APE) between the true pose value and the registration result. The whole process of the experiment is shown in Figure 1. All methods are implemented depending on the robot operating system (ROS) and the point cloud library (PCL).



Figure 1. The whole process of experiment.

## 4.3. Evaluation Indicators

4.3.1. RMSE

*RMSE* is a common method used to evaluate the accuracy of single-frame point cloud registration, in meters [81]. The *RMSE* calculation formula is as follows:

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^{m} \|(Rp_i + t) - q_i\|^2}$$
(7)

#### 4.3.2. ATE and RPE

In this paper, ATE and RPE [82] are used to evaluate the accuracy of continuous-frame registration. ATE is equivalent to the following APE, which is suitable for evaluating the overall performance of the registration algorithm.

RPE is used to describe the accuracy of the pose difference between two frames separated by a certain time difference, which is equivalent to the error of the direct measurement of the odometry. The standard is appropriate for estimating the drift of the system.

#### 4.4. Experimental Results

## 4.4.1. Single Frame Experimental Comparison

All of the above registration methods are required to set parameters, and different application scenarios need to set different parameters. The initial parameters are the default parameters set in PCL, and then the parameters are chosen according to the registration accuracy and time. First, we discuss the transformation epsilon (the maximum allowable difference between two consecutive transformations) and the maximum number of iterations. Furthermore, for the ICP series of algorithms, all filtered point clouds are used for registration. For point-to-plane ICP (PTPLICP, PTPLOMPICP), analyze the number of points using the calculated normal. For GICP, FastGICP, and FastVGICP, discuss the effect of using different numbers of points to calculate the covariance matrix on registration. For the NDT series algorithms, the relationship between voxel side length and registration time, and accuracy is analyzed.

Figure 2a–f shows the relationship between registration time and registration accuracy when different registration parameters are selected for each registration algorithm. The x-axis represents the registration time, the y-axis is the registration accuracy, and broken lines of different colors correspond to different registration algorithms. The values in the box above the broken line are registration parameter values, corresponding to the points on the broken line. The unselected parameters are represented by yellow boxes, and the parameters in the red box are the final parameters selected by the algorithm. In addition, point cloud registration is an error accumulation process. The frequency of the LiDAR used in the experiment is 10 Hz; that is, 600 frames of data are emitted per minute. To ensure that the error is within tens of centimeters, the *RMSE* error of a single frame should be retained to three decimal places, so the error level of the *RMSE* of 0.001 is reasonable.

Figure 2a shows the relationship between the registration time and the registration accuracy (*RMSE*) of the t transformation epsilon from  $1 \times 10^{-10}$  to  $1 \times 10^{-1}$ . Each registration algorithm is sensitive to the parameter transformation epsilon. The final value of the parameter transformation epsilon is shown in Figure 2b It can be seen from Figure 2c that as the number of iterations increases, the registration accuracy and registration time increase until convergence. When the maximum number of iterations is 10, all registration algorithms achieve convergence. Figure 2d shows the registration effect of the point-to-plane ICP algorithm when calculating normals with different points. PTPLICP and PTPLOMPICP achieve the best registration effect when calculating normals with 40 points. The number of points used by GICP, FastGICP, and FastVGICP to calculate the covariance matrix ranges from 10 to 100 intervals of 10. The covariance matrix is calculated with 40, 100, and 80 points, respectively. Registration accuracy is considered first, and then the registration time. After many experiments, we set the NDT series algorithm voxel side length range to 1.0 to 1.9 interval 0.1. It can be seen from Figure 2f that the registration effect of the NDT series algorithms changes obviously with the voxel side length. The voxel side lengths of NDT, NDTOMP1, and NDTOMP7 are 1.4, 1.8, and 1.8.



**Figure 2.** Result comparison of registration algorithms under different parameter settings. (a) Transformation epsilon comparison; (b) transformation epsilon result comparison; (c) maximum iterations comparison; (d) points for normals comparison; (e) points for covariances comparison; (f) NDT voxel comparison.

After configuring the parameters, the final registration results are shown in Figure 3. ICP and LMICP have the highest registration accuracy, but the speed is relatively slow, especially in LMICP. The registration speed of FastVGICP and NDTOMP1 is faster (less than 40 ms), but the accuracy is significantly lower than that of ICP. It can be seen from Figure 2a, c that ICP, LMICP, and NDT are more sensitive to parameters, especially ICP and LMICP, and that other registration performance changes are relatively small with parameters. Compared with GICP, FastGICP and FastVGICP have better stability (Figure 2e). It can be seen from Figure 2d that PTPLOMPICP has a significant improvement in registration speed compared with PTPLICP. NDTOMP1 has obvious speed advantages over NDT and NDTOMP7, and NDTOMP7 registration accuracy is more stable (Figure 2f).



Figure 3. Comparison of registration algorithm results.

## 4.4.2. Continuous Frame Experiment Comparison

In contrast to the single-frame experiment of scan-to-scan registration, in a continuousframe experiment, we use scan-to-map registration. The LiDAR frequency is 10 Hz, the IMU frequency is 100 Hz, and the GNSS data is about 2 Hz (uneven). In order to facilitate the performance comparison of the algorithm, after aligning the initial timestamp of the data, we use GNSS as a benchmark to select data with a time difference of less than 0.05 from the LiDAR data. Similarly, for the IMU data, we select the data with the smallest time difference between the IMU and the GNSS, and the time difference is less than 0.05. The data that meet the above conditions are used as the reference value to evaluate the performance of the algorithm.

Figure 4 is the trajectory comparison map of each registration algorithm, and the dotted line is the true value of the trajectory. From the diagram, it can be observed that the registration results of FastGICP, FastVGICP, PTPLICP, PTPLOMPICP, NDT, and NOTOMP7 are better, and the coincidence between the trajectory and the real value is higher. The trajectories of ICP, LMICP, GICP, and NDTOMP1 are quite different from the real values, and the registration results are poor, especially for ICP and GICP. Some trajectories of the ICP and GICP algorithms have exceeded the port range, so they are not shown. The registration results of the six algorithms, such as FastGICP, are close to the true values, but it can be observed from the local enlarged image that the registration results of some regions are also significantly different from the true values. In the following, we further compare and analyze the registration algorithms with better results. The PTPLICP and PTPLOMPICP trajectories are completely coincident, so only one of the two is discussed.

Figure 5 shows the comparison of the output trajectories of the registration algorithms in the x, y, and z directions. The trajectories of each registration algorithm in the x and y directions are under a high degree of coincidence with the true value of the trajectory, which proves that the registration algorithm has better registration results in the horizontal plane. It can be seen from the local graph that in the x and y directions, the algorithms with slightly worse registration results than other algorithms are FastGICP and PTPLOMPICP, respectively. The difference between the results in the z direction is obvious. Depending on the data analysis, the reason for this phenomenon should be that the observation information in the z direction is less, the observation incident angle in the z direction is large, and the precision is poor. FastGICP, FastVGICP, and PTPLOMPICP have better registration results in the z direction.

meters(m)

meters(m)





Figure 4. Comparison of output trajectories of registration algorithm.

It can be seen from Figure 6 and the local graph that each registration algorithm has a similar performance on the yaw angle. The performance of each algorithm is different in terms of roll angle and pitch angle. Compared with FastGICP, FastVGICP, PTPLICP, and PTPLOMPICP, NDT and NDTOMP7 are more unstable than the actual value. The reason why the roll angle is not much larger than the pitch angle is that the two sides of the ship are equipped with yellow, which can reduce the frequency of roll.



Figure 5. Comparison of output trajectories of registration algorithm in x, y, and z directions.



Figure 6. Comparison of output pose Euler angles of registration algorithm.

Figure 7 shows the roll and pitch error curves. For ease of viewing, the curves are smoothed by the Savitzky–Golay filter, where the window length and polynomial fitting order are 15 and 3, respectively. From the smoothed curve, it can be seen that the trend of roll and pitch is similar, and both contain multiple local gentle (small change range) and local steep (large change range) stages. Registration is an ongoing process of error accumulation. The gentle error curve indicates that the registration error is small and the registration result is good. On the contrary, the steep error curve indicates that the

registration error is enormous and the result is poor. The changing trend of roll and pitch is similar, so only one of them (roll) is discussed. We selected four cases for further analysis, which are case1 and case2 with obvious error changes, and case3 and case4 with gentle error changes. The visualization results of point cloud and image data are shown in Figures 8 and 9, respectively.



Figure 7. Roll and pitch errors of registration algorithm.



Figure 8. Top view of ASV point cloud in different periods.



Figure 9. Image data corresponding to point cloud.

Figure 8 is a 5-frame point cloud top view selected in the 1,2,3,4 parts of Figure 7. Figure 9 shows the corresponding image data, with an interval of 1 s between every two frames of data. The coordinate values contained in Figure 8 are the x and y coordinates of the current ship in the world coordinate system. It can be seen from the figure that case1 and case2 have obvious turns. In case3 and case4, the ship is closer to the straight line.

Table 1 is the mean ( $\mu$ ) and standard deviation ( $\sigma$ ) of the inter-frame distance of the four-part point cloud in Figure 8, which is the square of the distance between the corresponding points in the two frame point clouds. It can be seen from the data in the table that the  $\mu$  and  $\sigma$  of the point cloud distance in case1 and case2 are large, indicating that the distance between the corresponding points of the two frames of point clouds is large and discrete, and the distance between the corresponding points is large, which is consistent with the above ship-turning conclusion. The  $\mu$  and  $\sigma$  of the point cloud distance in case3 and case4 are small, so the distance and distance difference between the corresponding points of the point cloud are small, which also confirms that the ship is more likely to sail near the straight line.

Case	μ/σ	F1-F2	F2-F3	F3-F4	F4-F5
1	μ	3.29155	5.59733	5.90157	2.33137
	σ	18.6138	33.5996	34.2802	14.5016
2	μ	1.39014	2.39899	2.35713	2.19866
	σ	15.1237	24.6198	20.1098	19.4458
3	μ	0.252314	0.266688	0.541945	0.217449
	σ	3.85721	3.53274	10.5645	6.2222
4	μ	0.084555	0.085174	0.424403	0.15355
	σ	1.67186	2.38434	18.0619	2.93024

Table 1. Mean and standard deviation of point cloud frame distance.

Note: F1-F2 represents from Frame1 to Frame2.

Figures 10 and 11 are the ATE and RPE results of the trajectory translation error output by the FastGICP algorithm obtained by the evo tool, where the RPE interval is 10 pose points. In addition, the information provided by the tool also includes *RMSE*, mean, and median. The evo evaluation results of each registration algorithm are shown in Table 2. FastGICP, FastVGICP, and PTPLOMPICP have smaller ATEs, which is consistent with the conclusions obtained in Figure 5. Compared with ATE, the RPE results of each

algorithm are less different. The algorithm with the smallest RPE is FastVGICP, followed by NDTOMP7. FastVGICP achieves optimal values on both ATE and RPE. Through the above analysis, the registration accuracy of each registration algorithm, from high to low, is FastVGICP, FastGICP, PTPLOMPICP, NDTOMP7, and NDT.







Relative trajectory error of pose translation

Figure 11. Evo tool RPE evaluation results for FastGICP algorithm.

Evo Evaluation		FastGICP	FastVGICP	PTPLOMPICP NDT		NDTOMP7
APE	RMSE	2.199127	1.862248	2.677981	3.565329	3.394351
	Mean	1.934627	1.67735	2.207614	2.885509	2.754741
	Median	1.680057	1.511793	1.757098	2.44847	2.298347
RPE	RMSE	1.340868	1.242829	1.330651	1.299537	1.283601
	Mean	1.078858	0.982106	1.089126	1.013066	1.002319
	Median	0.861822	0.854125	0.928976	0.812974	0.796367

**Table 2.** The evaluation results of five algorithms using evo tool.

The two experiments in this paper are different, and the results are not the same. Compared with single-frame registration, continuous-frame registration is a process of multiple registrations, which has a cumulative error and is more complicated. It can be seen from the experimental results that the algorithm with better single-frame registration results is not satisfactory at continuous frames, such as ICP and LMICP. Similarly, algorithms with better continuous-frame registration may not be very prominent on a single frame, such as PTPLICP (PTPLOMP). Experimental results show that ICP and LMICP are suitable for single-frame registration, and FastVGICP and FastGICP are more suitable for continuousframe registration.

## 5. Discussion

#### 5.1. Accuracy and Real-Time of Scan Matching Algorithm

It can be observed in the single-frame experiment that the parameter setting of the registration algorithm directly affects the accuracy and speed of registration. This paper also sets some parameters taking into account these two aspects. The parameter selection in this paper gives priority to accuracy, followed by speed. However, if an embedded system with limited computing power is used, this will not be a good choice. This experiment uses a 16-line LiDAR, and the number of laser points is not very large. It can be observed in the final registration result map that the registration speed is fast. The registration time of the NDT algorithm is more than 100 ms, and other registration algorithms are less than 100 ms. Scan-to-map has higher accuracy than scan-to-scan, but the registration time also increases accordingly. Real-time registration can be obtained by properly adjusting the filtering parameters.

#### 5.2. Discuss the Scan-to-Scan and Scan-to-Map

From the above, single-frame point cloud registration using scan-to-scan registration and continuous-frame registration utilizing scan-to-map registration. The scan-to-scan is the registration of two adjacent frames. The advantage is that the calculation is small, and the disadvantage is that the error accumulation is large, especially for long distances. Scan-to-map is the current frame and keyframe subgraph matching. The advantage is high precision; error accumulation is small; the disadvantage is that the calculation is large. ASV requires accurate positioning and six degrees of freedom state estimation when performing tasks. Considering the number of LiDAR lines, this paper chooses the scan-to-map registration method.

## 5.3. Point Cloud Registration at Ship Turning

For point cloud registration, the larger the overlapping part of the point cloud, the better the registration result, and the smaller the overlapping part usually leads to registration failure. In Figures 4 and 7, we can find that there is a large positioning error and state estimation error at the turning point. The reason for this phenomenon is that the ship's heading angular velocity is large at the turning point, which will lead to a low overlap rate of the two adjacent frame point clouds, resulting in large registration errors and poor positioning accuracy. This is also one of the main reasons for the low positioning accuracy of scan-matching technology.

#### 5.4. Differences of Scan Matching at Sea and on Road

The research on scan matching is mostly based on the land, such as wheeled robots, cars, and so on. Compared with the motion on land, the ASV sailing at sea is more violently affected by wind and waves, the motion is more complex, and it is more difficult to achieve accurate registration. In addition, the laser of the LiDAR hits the water surface, most of which is absorbed by water, and a small part of the mirror is reflected without an echo. Therefore, the laser point incident on the sea surface is almost not reflected back to the LiDAR. Compared with the LiDAR data on land, the point clouds scanned by the LiDAR

on the sea have no land information, and the number and characteristics of the point clouds are smaller, which further increases the difficulty of registration.

#### 6. Conclusions

This paper mainly studies the ASV maritime positioning and state estimation based on LiDAR, where the poses are obtained by scan matching. We introduced ICP, NDT, and their variants in detail and selected 10 different registration methods for testing and evaluation, including ICP, LMICP, GICP, FastGICP, FastVGICP, and PTPLICP, PTPLOMPICP, NDT, NDTOMP1, and NDTOMP7. The performance test experiment of the algorithm is divided into single-frame experiments and continuous-frame experiments. Single-frame experiments used scan-to-map registration. The accuracy of the single-frame experiment is evaluated by the *RMSE*, and the parameters of the algorithm are selected at the same time. The continuous-frame experiment uses the registration output overall trajectory, XYZ direction trajectory, Euler angle, ATE, and RPE to evaluate each registration algorithm.

The experimental results show that the single-frame registration ICP and LMICP have high accuracy, and the mean values of *RMSE* on 10 pairs of data are 0.965926 and 0.966460, respectively. NDTOMP1 and VGICP are fast in registration, and the average registration time is 19 ms and 31.4 ms, but the registration accuracy is much lower than ICP. For continuous-frame registration, the registration results of ICP, LMICP, GICP, and NDTOMP1 differ greatly from the truth values, especially ICP and GICP. VGICP and multi-threaded optimization GICP have the best effect, and the *RMSE* of ATE is 1.862248 and 2.199127, respectively. Compared with straight-line navigation, the distance between the corresponding points of two adjacent frames of point clouds increases and becomes more discrete when USV turns, and the registration accuracy is worse.

However, there are still some defects in this paper owing to the limitations of experimental conditions. The mechanism of loss of registration accuracy at turning could be revealed by more abundant data sets, such as different maneuvers, ASVs of different sizes, and LiDAR of different types. In addition, the influence of dynamic objects from a busy port on point cloud registration is the focus of our future study.

**Author Contributions:** Writing—original draft preparation, H.W.; writing—review and editing, Y.Y. and Q.J.; methodology, H.W., Y.Y. and Q.J.; software, H.W. and Q.J.; formal analysis, H.W.; validation, H.W.; investigation, H.W. and Q.J.; resources, Y.Y. and Q.J.; data curation, H.W., Y.Y. and Q.J.; visualization, H.W.; funding acquisition, Y.Y. and Q.J.; supervision, Y.Y. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China, grant number 52071049; the Doctoral Scientific Research Foundation of Liaoning Province of China, grant number 2020-BS-070; Ministry of Industry and Information Technology Letter "Intelligent Ship Comprehensive Test and Verification Research", grant number No. 2018(473); the Fundamental Research Funds for the Central Universities, grant number 3132022131, and the 2022 Liaoning Provincial Science and Technology Plan (Key) Project: R&D and Application of Autonomous Navigation System for Smart Ships in Complex Waters, grant number 2022JH1/10800096.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

# References

- Peng, Z.; Wang, J.; Wang, D.; Han, Q.-L. An Overview of Recent Advances in Coordinated Control of Multiple Autonomous Surface Vehicles. *IEEE Trans. Ind. Inform.* 2021, 17, 732–745. [CrossRef]
- 2. Bejarano, G.; N-Yo, S.; Orihuela, L. Velocity and Disturbance Robust Nonlinear Estimator for Autonomous Surface Vehicles Equipped with Position Sensors. *IEEE Trans. Control. Syst. Technol.* **2022**, *30*, 2235–2242. [CrossRef]

- 3. Luis, S.Y.; Reina, D.G.; Marín, S.L.T. A Multiagent Deep Reinforcement Learning Approach for Path Planning in Autonomous Surface Vehicles: The Ypacaraí Lake Patrolling Case. *IEEE Access* **2021**, *9*, 17084–17099. [CrossRef]
- 4. Liu, Z.; Zhang, Y.; Yu, X.; Yuan, C. Unmanned Surface Vehicles: An Overview of Developments and Challenges. *Annu. Rev. Control.* **2016**, *41*, 71–93. [CrossRef]
- 5. Jiang, X.; Xia, G. Sliding Mode Formation Control of Leaderless Unmanned Surface Vehicles with Environmental Disturbances. Ocean. Eng. 2022, 244, 110301. [CrossRef]
- 6. Jones, D.O.B.; Gates, A.R.; Huvenne, V.A.I.; Phillips, A.B.; Bett, B.J. Autonomous Marine Environmental Monitoring: Application in Decommissioned Oil Fields. *Sci. Total Environ.* **2019**, *668*, 835–853. [CrossRef]
- Luis, S.Y.; Reina, D.G.; Marin, S.L.T. A Deep Reinforcement Learning Approach for the Patrolling Problem of Water Resources Through Autonomous Surface Vehicles: The Ypacarai Lake Case. *IEEE Access* 2020, *8*, 204076–204093. [CrossRef]
- Wang, Y.-L.; Han, Q.-L. Network-Based Modelling and Dynamic Output Feedback Control for Unmanned Marine Vehicles in Network Environments. *Automatica* 2018, 91, 43–53. [CrossRef]
- Zhang, J.; Xiong, J.; Zhang, G.; Gu, F.; He, Y. Flooding Disaster Oriented USV & UAV System Development & Demonstration. In Proceedings of the OCEANS 2016, Shanghai, China, 19–22 September 2016; pp. 1–4.
- 10. Rätsep, M.; Parnell, K.E.; Soomere, T.; Kruusmaa, M.; Ristolainen, A.; Tuhtan, J.A. Surface Vessel Localization from Wake Measurements Using an Array of Pressure Sensors in the Littoral Zone. *Ocean. Eng.* **2021**, 233, 109156. [CrossRef]
- 11. Ng, H.-F.; Zhang, G.; Yang, K.-Y.; Yang, S.-X.; Hsu, L.-T. Improved Weighting Scheme Using Consumer-Level GNSS L5/E5a/B2a Pseudorange Measurements in the Urban Area. *Adv. Space Res.* **2020**, *66*, 1647–1658. [CrossRef]
- 12. Zhu, N.; Marais, J.; Bétaille, D.; Berbineau, M. GNSS Position Integrity in Urban Environments: A Review of Literature. *IEEE Trans. Intell. Transp. Syst.* 2018, 19, 2762–2778. [CrossRef]
- Al Borno, M.; O'Day, J.; Ibarra, V.; Dunne, J.; Seth, A.; Habib, A.; Ong, C.; Hicks, J.; Uhlrich, S.; Delp, S. OpenSense: An Open-Source Toolbox for Inertial-Measurement-Unit-Based Measurement of Lower Extremity Kinematics over Long Durations. *J. NeuroEngineering Rehabil.* 2022, 19, 22. [CrossRef] [PubMed]
- 14. Bonin-Font, F.; Burguera, A. Towards Multi-Robot Visual Graph-SLAM for Autonomous Marine Vehicles. *JMSE* 2020, *8*, 437. [CrossRef]
- 15. Callmer, J.; Törnqvist, D.; Gustafsson, F.; Svensson, H.; Carlbom, P. Radar SLAM Using Visual Features. *EURASIP J. Adv. Signal Process.* 2011, 2011, 71. [CrossRef]
- Wang, Z.; Zhang, Y. Estimation of Ship Berthing Parameters Based on Multi-LiDAR and MMW Radar Data Fusion. *Ocean. Eng.* 2022, 266, 113155. [CrossRef]
- 17. Li, C.; Dai, B.; Wu, T. Vision-Based Precision Vehicle Localization in Urban Environments. In Proceedings of the 2013 Chinese Automation Congress, Changsha, China, 7–8 November 2013; pp. 599–604.
- 18. Vivet, D.; Gérossier, F.; Checchin, P.; Trassoudaine, L.; Chapuis, R. Mobile Ground-Based Radar Sensor for Localization and Mapping: An Evaluation of Two Approaches. *Int. J. Adv. Robot. Syst.* **2013**, *10*, 307. [CrossRef]
- 19. Taketomi, T.; Uchiyama, H.; Ikeda, S. Visual SLAM Algorithms: A Survey from 2010 to 2016. *IPSJ Trans. Comput. Vis. Appl.* 2017, 9, 16. [CrossRef]
- 20. Sugiura, K.; Matsutani, H. A Universal LiDAR SLAM Accelerator System on Low-Cost FPGA. *IEEE Access* 2022, 10, 26931–26947. [CrossRef]
- 21. Kumar, G.A.; Patil, A.K.; Patil, R.; Park, S.S.; Chai, Y.H. A LiDAR and IMU Integrated Indoor Navigation System for UAVs and Its Application in Real-Time Pipeline Classification. *Sensors* 2017, *17*, 1268. [CrossRef]
- 22. Gao, Y.; Liu, S.; Atia, M.M.; Noureldin, A. INS/GPS/LiDAR Integrated Navigation System for Urban and Indoor Environments Using Hybrid Scan Matching Algorithm. *Sensors* 2015, *15*, 23286–23302. [CrossRef]
- 23. Lu, X.; Li, Y. Motion Pose Estimation of Inshore Ships Based on Point Cloud. Measurement 2022, 205, 112189. [CrossRef]
- 24. Besl, P.J.; McKay, N.D. Method for Registration of 3-D Shapes. In Proceedings of the Sensor Fusion IV: Control Paradigms and Data Structures, Boston, MA, USA, 12–15 November 1992; SPIE: Bellingham, WA, USA, 1992; Volume 1611, pp. 586–606.
- Biber, P.; Strasser, W. The Normal Distributions Transform: A New Approach to Laser Scan Matching. In Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453), Las Vegas, NV, USA, 27–31 October 2003; Volume 3, pp. 2743–2748.
- Ren, R.; Fu, H.; Xue, H.; Li, X.; Hu, X.; Wu, M. LiDAR-based Robust Localization for Field Autonomous Vehicles in Off-road Environments. J. Field Robot. 2021, 38, 1059–1077. [CrossRef]
- Ji, Z.; Singh, S. LOAM: Lidar Odometry and Mapping in Real-Time. In Proceedings of the Robotics: Science and Systems Conference, Berkeley, CA, USA, 12–16 July 2014; Volume 2, pp. 1–9.
- Rusu, R.B.; Bradski, G.; Thibaux, R.; Hsu, J. Fast 3D Recognition and Pose Using the Viewpoint Feature Histogram. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, China, 18–22 October 2010; pp. 2155–2162.
- Rusu, R.B.; Blodow, N.; Beetz, M. Fast Point Feature Histograms (FPFH) for 3D Registration. In Proceedings of the 2009 IEEE International Conference on Robotics and Automation, Kobe, Japan, 12–17 May 2009; pp. 3212–3217.
- You, B.; Chen, H.; Li, J.; Li, C.; Chen, H. Fast Point Cloud Registration Algorithm Based on 3DNPFH Descriptor. *Photonics* 2022, 9, 414. [CrossRef]

- 31. Zováthi, Ö.; Nagy, B.; Benedek, C. Point Cloud Registration and Change Detection in Urban Environment Using an Onboard Lidar Sensor and MLS Reference Data. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *110*, 102767. [CrossRef]
- Li, L.; Wang, R.; Zhang, X. A Tutorial Review on Point Cloud Registrations: Principle, Classification, Comparison, and Technology Challenges. *Math. Probl. Eng.* 2021, 2021, e9953910. [CrossRef]
- Bash, E.A.; Wecker, L.; Rahman, M.M.; Dow, C.F.; McDermid, G.; Samavati, F.F.; Whitehead, K.; Moorman, B.J.; Medrzycka, D.; Copland, L. A Multi-Resolution Approach to Point Cloud Registration without Control Points. *Remote Sens.* 2023, 15, 1161. [CrossRef]
- 34. Guevara, D.J.; Gené-Mola, J.; Gregorio, E.; Torres-Torriti, M.; Reina, G.; Cheein, F.A.A. Comparison of 3D Scan Matching Techniques for Autonomous Robot Navigation in Urban and Agricultural Environments. *JARS* **2021**, *15*, 024508. [CrossRef]
- 35. Wang, G.; Wu, X.; Jiang, S.; Liu, Z.; Wang, H. Efficient 3D Deep LiDAR Odometry. *IEEE Trans. Pattern Anal. Mach. Intell.* 2022, 14, 1–17. [CrossRef]
- Clunie, T.; DeFilippo, M.; Sacarny, M.; Robinette, P. Development of a Perception System for an Autonomous Surface Vehicle Using Monocular Camera, LIDAR, and Marine RADAR. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 14112–14119.
- Jing, Q.; Wang, H.; Hu, B.; Liu, X.; Yin, Y. A Universal Simulation Framework of Shipborne Inertial Sensors Based on the Ship Motion Model and Robot Operating System. *J. Mar. Sci. Eng.* 2021, *9*, 900. [CrossRef]
- Liu, H.; Liu, T.; Li, Y.; Xi, M.; Li, T.; Wang, Y. Point Cloud Registration Based on MCMC-SA ICP Algorithm. *IEEE Access* 2019, 7, 73637–73648. [CrossRef]
- 39. Li, P.; Wang, R.; Wang, Y.; Tao, W. Evaluation of the ICP Algorithm in 3D Point Cloud Registration. *IEEE Access* 2020, *8*, 68030–68048. [CrossRef]
- 40. Lu, J.; Wang, W.; Shao, H.; Su, L. Point Cloud Registration Algorithm Fusing of Super 4PCS and ICP Based on the Key Points. In Proceedings of the 2019 Chinese Control Conference (CCC), Guangzhou, China, 27–30 July 2019; pp. 4439–4444.
- 41. Shi, X.; Liu, T.; Han, X. Improved Iterative Closest Point(ICP) 3D Point Cloud Registration Algorithm Based on Point Cloud Filtering and Adaptive Fireworks for Coarse Registration. *Int. J. Remote Sens.* **2020**, *41*, 3197–3220. [CrossRef]
- 42. Censi, A. An ICP Variant Using a Point-to-Line Metric. In Proceedings of the 2008 IEEE International Conference on Robotics and Automation, Pasadena, CA, USA, 19–23 May 2008; pp. 19–25.
- 43. Chen, Y.; Medioni, G. Object Modelling by Registration of Multiple Range Images. *Image Vis. Comput.* **1992**, *10*, 145–155. [CrossRef]
- 44. Segal, A.; Hhnel, D.; Thrun, S. Generalized-ICP. In Proceedings of the Robotics: Science and Systems V, University of Washington, Seattle, WA, USA, 28 June–1 July 2009; Volume 2, p. 435.
- Koide, K.; Yokozuka, M.; Oishi, S.; Banno, A. Voxelized GICP for Fast and Accurate 3D Point Cloud Registration. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 11054–11059.
- Sharp, G.C.; Lee, S.W.; Wehe, D.K. Invariant Features and the Registration of Rigid Bodies. In Proceedings of the 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C), Detroit, MI, USA, 10–15 May 1999; Volume 2, pp. 932–937.
- Minguez, J.; Montesano, L.; Lamiraux, F. Metric-Based Iterative Closest Point Scan Matching for Sensor Displacement Estimation. IEEE Trans. Robot. 2006, 22, 1047–1054. [CrossRef]
- Serafin, J.; Grisetti, G. NICP: Dense Normal Based Point Cloud Registration. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 742–749.
- Granger, S.; Pennec, X. Multi-Scale EM-ICP: A Fast and Robust Approach for Surface Registration. In Computer Vision, Proceedings of the ECCV 2002: 7th European Conference on Computer Vision, Copenhagen, Denmark, 28–31 May 2002; Heyden, A., Sparr, G., Nielsen, M., Johansen, P., Eds.; Springer: Berlin/Heidelberg, Germany, 2002; pp. 418–432.
- 50. Hong, S.; Ko, H.; Kim, J. VICP: Velocity Updating Iterative Closest Point Algorithm. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–7 May 2010; pp. 1893–1898.
- Alismail, H.; Baker, L.D.; Browning, B. Continuous Trajectory Estimation for 3D SLAM from Actuated Lidar. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 6096–6101.
- 52. Yang, J.; Li, H.; Campbell, D.; Jia, Y. Go-ICP: A Globally Optimal Solution to 3D ICP Point-Set Registration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 2241–2254. [CrossRef]
- 53. Han, J.; Yin, P.; He, Y.; Gu, F. Enhanced ICP for the Registration of Large-Scale 3D Environment Models: An Experimental Study. *Sensors* 2016, 16, 228. [CrossRef] [PubMed]
- 54. Greenspan, M.; Yurick, M. Approximate K-d Tree Search for Efficient ICP. In Proceedings of the Fourth International Conference on 3-D Digital Imaging and Modeling, Banff, AB, Canada, 6–10 October 2003; pp. 442–448.
- 55. Nuchter, A.; Lingemann, K.; Hertzberg, J. Cached K-d Tree Search for ICP Algorithms. In Proceedings of the Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007), Montreal, QC, Canada, 21–23 August 2007; pp. 419–426.
- 56. Fitzgibbon, A.W. Robust Registration of 2D and 3D Point Sets. *Image Vis. Comput.* 2003, 21, 1145–1153. [CrossRef]
- Pavlov, A.L.; Ovchinnikov, G.W.; 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, QLD, Australia, 21–25 May 2018; pp. 3407–3412.

- Zhang, J.; Yao, Y.; Deng, B. Fast and Robust Iterative Closest Point. *IEEE Trans. Pattern Anal. Mach. Intell.* 2022, 44, 3450–3466.
   [CrossRef]
- 59. Bouaziz, S.; Tagliasacchi, A.; Pauly, M. Sparse Iterative Closest Point. Comput. Graph. Forum 2013, 32, 113–123. [CrossRef]
- Agamennoni, G.; Fontana, S.; Siegwart, R.Y.; Sorrenti, D.G. Point Clouds Registration with Probabilistic Data Association. In Proceedings of the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems, Daejeon, Republic of Korea, 9–14 October 2016.
- 61. Dong, J.; Peng, Y.; Ying, S.; Hu, Z. LieTrICP: An Improvement of Trimmed Iterative Closest Point Algorithm. *Neurocomputing* **2014**, 140, 67–76. [CrossRef]
- 62. Rusinkiewicz, S. A Symmetric Objective Function for ICP. ACM Trans. Graph. 2019, 38, 85. [CrossRef]
- 63. Du, S.; Xu, G.; Zhang, S.; Zhang, X.; Gao, Y.; Chen, B. Robust Rigid Registration Algorithm Based on Pointwise Correspondence and Correntropy. *Pattern Recognit. Lett.* **2020**, *132*, 91–98. [CrossRef]
- 64. Wu, Z.; Chen, H.; Du, S.; Fu, M.; Zhou, N.; Zheng, N. Correntropy Based Scale ICP Algorithm for Robust Point Set Registration. *Pattern Recognit.* **2019**, *93*, 14–24. [CrossRef]
- Dellenbach, P.; Deschaud, J.-E.; Jacquet, B.; Goulette, F. CT-ICP: Real-Time Elastic LiDAR Odometry with Loop Closure. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 23–27 May 2022; pp. 5580–5586.
- Magnusson, M.; Lilienthal, A.; Duckett, T. Scan Registration for Autonomous Mining Vehicles Using 3D-NDT. J. Field Robot. 2007, 24, 803–827. [CrossRef]
- Magnusson, M.; Nuchter, A.; Lorken, C.; Lilienthal, A.J.; Hertzberg, J. Evaluation of 3D Registration Reliability and Speed—A Comparison of ICP and NDT. In Proceedings of the 2009 IEEE International Conference on Robotics and Automation, Kobe, Japan, 12–17 May 2009; pp. 3907–3912.
- 68. Koide, K.; Miura, J.; Menegatti, E. A Portable Three-Dimensional LIDAR-Based System for Long-Term and Wide-Area People Behavior Measurement. *Int. J. Adv. Robot. Syst.* **2019**, *16*, 172988141984153. [CrossRef]
- Zaganidis, A.; Magnusson, M.; Duckett, T.; Cielniak, G. Semantic-Assisted 3D Normal Distributions Transform for Scan Registration in Environments with Limited Structure. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 4064–4069.
- Jun, L.; Wei, L.; Donglai, D.; Qiang, S. Point Cloud Registration Algorithm Based on NDT with Variable Size Voxel. In Proceedings of the 2015 34th Chinese Control Conference (CCC), Hangzhou, China, 28–30 July 2015; pp. 3707–3712.
- Hong, H.; Lee, B. A Method of Generating Multi-Scale Disc-Like Distributions for NDT Registration Algorithm. Int. J. Mech. Eng. Robot. Res. 2016, 5, 52. [CrossRef]
- 72. Ulas, C.; Temeltas, H. A 3D Scan Matching Method Based on Multi-Layered Normal Distribution Transform. *IFAC Proc. Vol.* **2011**, 44, 11602–11607. [CrossRef]
- Ulas, C.; Temeltas, H. A Fast and Robust Feature-Based Scan-Matching Method in 3D SLAM and the Effect of Sampling Strategies. Int. J. Adv. Robot. Syst. 2013, 10, 396. [CrossRef]
- Hong, H.; Lee, B.H. Key-Layered Normal Distributions Transform for Point Cloud Registration. *Electron. Lett.* 2015, 51, 1986–1988. [CrossRef]
- Hong, H.; Lee, B.H. Probabilistic Normal Distributions Transform Representation for Accurate 3D Point Cloud Registration. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 3333–3338.
- 76. Zaganidis, A.; Zerntev, A.; Duckett, T.; Cielniak, G. Semantically Assisted Loop Closure in SLAM Using NDT Histograms. In Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 4562–4568.
- 77. Lee, S.; Kim, C.; Cho, S.; Myoungho, S.; Jo, K. Robust 3-Dimension Point Cloud Mapping in Dynamic Environment Using Point-Wise Static Probability-Based NDT Scan-Matching. *IEEE Access* **2020**, *8*, 175563–175575. [CrossRef]
- 78. Deng, Q.; Sun, H.; Chen, F.; Shu, Y.; Wang, H.; Ha, Y. An Optimized FPGA-Based Real-Time NDT for 3D-LiDAR Localization in Smart Vehicles. *IEEE Trans. Circuits Syst. II Express Briefs* **2021**, *68*, 3167–3171. [CrossRef]
- 79. Zhong, K.; Liu, Y.; Yang, J.; Lu, M.; Zhang, J. A Factor Graph Optimization Mapping Based on Normaldistributions Transform. *Turk. J. Electr. Eng. Comput. Sci.* 2022, 30, 1127–1141. [CrossRef]
- Hu, B.; Liu, X.; Jing, Q.; Lyu, H.; Yin, Y. Estimation of Berthing State of Maritime Autonomous Surface Ships Based on 3D LiDAR. Ocean. Eng. 2022, 251, 111131. [CrossRef]
- Yang, J.; Cao, Z.; Zhang, Q. A Fast and Robust Local Descriptor for 3D Point Cloud Registration. *Inf. Sci.* 2016, 346, 163–179. [CrossRef]
- Sturm, J.; Engelhard, N.; Endres, F.; Burgard, W.; Cremers, D. A Benchmark for the Evaluation of RGB-D SLAM Systems. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 573–580.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.