Article

FFT-Based Scan-Matching for SLAM Applications with Low-Cost Laser Range Finders

Guolai Jiang ^{1,2,3}, Lei Yin ¹, Guodong Liu ^{2,3}, Weina Xi ¹ and Yongsheng Ou ^{1,2,4,*}

¹ Guangdong Provincial Key Laboratory of Robotics and Intelligent System, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China; gl.jiang@siat.ac.cn (G.J.); lei.yin@siat.ac.cn (L.Y.); wn.xi@siat.ac.cn (W.X.)

² Shenzhen College of Advanced Technology, University of Chinese Academy of Sciences, Guangdong 518055, China; gd.liu@siat.ac.cn (G.L.)

³ Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Guangdong 518055, China

⁴ The CAS Key Laboratory of Human-Machine-Intelligence Synergic Systems, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Guangdong 518055, China

* Correspondence: ys.ou@siat.ac.cn

Received: 22 November 2018; Accepted: 18 December 2018; Published: 22 December 2018

Abstract: Simultaneous Localization and Mapping (SLAM) is an active area of robot research. SLAM with a laser range finder (LRF) is effective for localization and navigation. However, commercial robots usually have to use low-cost LRF sensors, which result in lower resolution and higher noise. Traditional scan-matching algorithms may often fail while the robot is running too quickly in complex environments. In order to enhance the stability of matching in the case of large pose differences, this paper proposes a new method of scan-matching mainly based on Fast Fourier Transform (FFT) as well as its application with a low-cost LRF sensor. In our method, we change the scan data within a range of distances from the laser to various images. FFT is applied to the images to determine the rotation angle and translation parameters. Meanwhile, a new kind of feature based on missing data is proposed to determine the rough estimation of the rotation angle under some representative scenes, such as corridors. Finally, Iterative Closest Point (ICP) is applied to determine the best match. Experimental results show that the proposed method can improve the scan-matching and SLAM performance for low-cost LRFs in complex environments.

Keywords: Mobile robot; SLAM; low-cost; laser range finder; Fast Fourier Transform

1. Introduction

With the rapid development of artificial intelligence and pattern recognition technology, intelligent robots have entered all aspects of industrial automation and human life. One of the most important features/functions of robot intelligence is to move autonomously, which is usually supported by precise localization and reliable navigation. To achieve this, the common technique is Simultaneous Localization and Mapping (SLAM). SLAM includes two parts: localization and mapping, where localization is an important premise of mapping. The SLAM problem can be solved by maintaining a robot-state (pose and map) probability-density function or its moments [1]. Depending on whether the environment is pre-given, localization can be divided into localization in a known environment and localization in an unknown environment [2]. The localization in a known environment mainly considers the problem of positioning accuracy. Localization in an unknown environment often requires the use of external sensors to obtain information for estimating the robot's position after processing. The self-localization method can also be divided into two categories: relative positioning and absolute positioning. Relative positioning refers to determining the relative distance and direction in which a robot moved in a short

time. This method measures the change of the mobile robot's current location related to its initial position, usually based on dead reckoning, Kalman Filter, Markov, Montecarlo, and Extended Kalman Filter (EKF). Absolute positioning refers to the direct determination of position in the world coordinate system that mainly includes map matching and positioning [3].

When a robot is moving, the position calculated by the internal sensors (such as an encoder, Inertial Measurement Unit (IMU), etc.) inevitably drifts. As a result, sensors that can perceive the surrounding environment are usually needed for robot positioning, including vision sensors, range sensors, GPS, etc. Vision-based approaches have been widely studied in recent years [4]. They analyze and track natural landmarks [5], significant features [6,7], or image pixel values directly [8–10]. Special cameras are used for achieving better performance under different situations, such as omnidirectional camera approaches [11,12], RGB-D camera approaches [10,13], etc. However, vision methods usually have high computational complexity or it is difficult to create maps for navigation and obstacle avoidance. Laser range finder (LRF) sensors occupy an essential role in robot navigation because they can build occupancy grid maps with low computational cost.

SLAM based on LRF sensors has been studied thoroughly [14–20]. Nüchter [14] presented an approach to parallelize two well-known mapping algorithms, namely Iterative Closest Point (ICP) and Lu Milios (LUM), which were used for scan matching in Robotic 3D mapping. Wolcott and Eustice [15] achieved real-time performance by developing a novel branch-and-bound, multi-resolution approach that made use of rasterized lookup tables of Gaussian mixture maps. Methods based on particle filters were proven to be effective for building the map based on the odometry message and scan data [16,17]. Grisetti et al. [16] proposed a SLAM method based on an improved Rao-Blackwellized particle filter, which used proposal distribution and an adaptive resampling technique to improve accuracy and reduce computational resource consumption. Particle distribution not only considers the motion model, but also considers observations. The advantage of this method is that it avoids the nonlinear problem. The problem is that as more particles are needed, more computing resources are consumed. Steux and Hamzaoui [17] implemented a minimal SLAM system in 2010 using 200 lines of C code, called Core-SLAM. The algorithm is simplified into two processes of distance calculation and map update, which greatly reduces the consumption of computing resources. However, its particle filter and matching method are too simple, which could lead to an erroneous position estimation, according to [18]. The graph optimization used to be a common method in visual SLAM, and it was also introduced into LRF-based SLAM. In 2010, Konolige et al. [19] proposed a SLAM that uses graph optimization instead of particle filters. This method uses Sparse point adjustment to solve the problem of matrix solution in nonlinear optimization. Cartographer [20] is another new SLAM solution proposed by Google. It adopts the framework of graph optimization and adds the concept of a submap to realize laser loop closure. For robot systems, the use of multi-sensor fusion for mapping and localization is common [21–25]. Two or more sensors, such as lidar, camera, IMU, encoder, etc., could be combined to achieve a more robust performance of SLAM [16,20–25] by using EKF, Unscented Kalman Filter (UKF), graph optimization, and so on.

One of the most heavily relied-upon tools for LRF-based SLAM is scan-matching; it is described as the problem of registering two laser scans in order to determine the relative positions from which the scans were obtained. The traditional method for scan matching is the Iterative Closest Point (ICP) algorithm proposed by Besl and McKay [26] in 1992. On account of its simplicity and low complexity, the ICP algorithm used to be very popular. Although it is commonly used, ICP has some defects, such as falling into local extremum, depending on initial registration, and having a high repetition rate. Lu and Milios [27] describe two scan-matching methods based on ICP. The first method considers the rotational and translational components separately: alternately fixing one, then optimizing the other. Given rotation, least-squares is used for optimizing translation. Feature-based methods were then introduced by Bosse [28] and Olson [29]. They extract pairs of points that are likely to be part of a connected surface, or lines/splines first, then scan points are matched through these features. Lv et al. [30] proposed a method that uses the Radial distance and Incident Angle (RIA) for scan

alignment, which can efficiently estimate the relative transformation between two scans. There are also many other scan-matching methods, such as using polar coordinates [31], Hough transform [32], histogram [33,34], correlation cost function [35,36], and Normal Distributions Transform (NDT) [37], etc. A study of perfect match [38] and its comparison to different methods is given in [39].

For commercial robotics, the cost of traditional LRF sensors are usually unacceptable. Serial kinds of low-cost LRF sensors (price lower than 600 RMB) have been developed in recent years, such as Rplidar by the Slamtec Company and N1-360 by the Leishen Company, etc. Compared with common expensive LRF sensors, low-cost LRF sensors have much poorer performance in range, precision, and resolution. As a result, scan matching may often fail while using low-cost LRF sensors, especially when the robot is running too quickly in complex environments, and it may cause failure to mapping and localization. High performance and low cost cannot be balanced, but it is an important concern for the realization of a commercial robot SLAM. For the commercial application of robotics, how to improve the SLAM algorithms so that they can adapt to low-cost and low-performance sensors is now becoming a key problem.

Most LRF-based SLAM methods need a good initial guess for scan-matching in order to diminish the range of searching and speed up the process. Odometry message is a popular choice [16,17] for that initial guess. Low-cost LRF sensors have much lower detection frequency and lower resolution, it may make the scan matching fail between two adjacent frames, especially when the odometry drifts.

If the scan data is considered as an image, which indicates the outline of obstacles near the robot, then the scan-matching problem can be transformed to an image registration problem. There are many image registration approaches that can match two positions in a long distance without any good initial guess for searching. However, scan images are quite different from normal images, since they have sparse points collected by lidar. As a result, normal feature-based image registration methods, such as Scale-Invariant Feature Transform (SIFT) [40], Speeded-Up Robust Features (SURF) [41], ORiented Brief (ORB) [42], etc., may not be suitable. We need to find the shift of the image by considering all the data in the image. By changing the image to the frequency domain, the FFT transform is suitable for this step. The idea in this paper is mainly derived from FFT-based image registration [43–46], because the scan image is simple and lies in one plane.

In our previous works [47,48], we proposed the FFT-based scan-matching algorithm and its improvement FFT-ICP. In this paper, we detailed previous work about data pretreatment. To address the low detection range problem of low-cost LRF sensors, we added a pre-alignment module based on missing data features to determine the rough rotation angle before FFT processing under certain conditions. We also propose a new solution for FFT-ICP scan-matching with low-cost LRF sensors.

The rest of the paper is organized as follows: In Section 2, we propose robot modeling and data pretreatment, as well as missing data feature extracting and matching; In Section 3, the FFT-based scan-matching algorithm is introduced; In Section 4, the complete solution of the improved FFT-ICP scan-matching is proposed; In Section 5, by using a cleaning robot with only a low-cost LRF sensor in different complex indoor environments, experiments are completed with the proposed method and comparative methods. Section 6 outlines our conclusions.

2. Modeling, Data Pretreatment, and Missing Data Features

2.1. Robot Modeling

In our work, a two-wheel driven robot moving on an indoor flat plane was studied, as shown in Figure 1. The scan-matching problem here is defined as finding the transformation between two robot poses based on the scan data acquired on each pose. Two coordinate systems are established here, marked as model coordinate system ($X_M O_M Y_M$) and scene coordinate system ($X_S O_S Y_S$). The transformation parameters are X , Y , and α , where X and Y are the translation parameters between the two poses and α is the rotation angle.

Figure 1. Robot modeling and different poses.

The rotation matrix R and translation matrix T can be used to represent the transformation between the two poses:

$$R = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \quad (1)$$

$$T = \begin{bmatrix} X \\ Y \end{bmatrix} \quad (2)$$

2.2. Scan Data Pretreatment

In our work, the scan-matching problem is transformed to an image registration problem. As a result, the scan data must be changed to an image. The raw data of one scan collected by the LRF sensor can be represented as

$$Z = \{(\theta_i, d_i)\} \quad (3)$$

where (θ_i, d_i) is a scan point detected on the direction angle of θ_i with distance d_i . One scan contains n scan points in total. If the scan point is invalid, distance $d_i = 0$. Invalid points usually exist in the directions where obstacles are out of the measurement range. When using a low-cost LRF sensor with limited detection range, such invalid points usually occur in the scenes of corridor, hall, etc.

The image size has a great influence on the amount of computation. Fortunately, the effective detection range of a low-cost LRF sensor is limited. As a result, this paper chose the data within a certain distance from the robot ($0 < d_i \leq d_{\max}$) as the data source for subsequent calculations.

The corresponding position (x_i, y_i) of scan point (θ_i, d_i) in the image is:

$$\begin{cases} x_i = \mu d_i \cos \theta_i + w/2 \\ y_i = \mu d_i \sin \theta_i + h/2 \end{cases} \quad (4)$$

where w and h are the width and height of the image and μ is the factor for transforming actual distance to pixel distance. A binary image with values on corresponding positions (x_i, y_i) set to 1 (stands for obstacles) and others set to 0 (stands for free) is generated. An example for data of the scan-image is shown in Figure 2.

Figure 2. (a) Simulation of scan in environment, (b) raw scan data, and (c) scan image.

2.3. The Missing Data Features

The raw laser data collected by LRF sensors usually contain missing data caused by out of range or unreliable measurements. Such missing data occurs more often for a low-cost LRF sensor because of the low detection range. To the best of our knowledge, almost all previous related works ignored or threw away the missing data because they assumed the laser sensor was perfect or the missing data were meaningless.

However, such missing data is meaningful. For a robot with a low-cost LRF sensor running in normal indoor environments without floor glasses, the fail detection on certain direction angles usually indicates an out of measurement range in that direction. In fact, some navigation applications interpolate such missing data to the maximum distance value to make the running of the robot more robust in practice.

In this paper, we put forward a new approach for making use of such missing data with ($d_i = 0$) or $d_i > d_{\max}$, called missing data feature. For raw scan data, the missing data feature is extracted as

$$f_{\text{mis}} = \{(\beta_1, \varphi_1), (\beta_2, \varphi_2), \dots, (\beta_m, \varphi_m)\} \quad (5)$$

An angle range of continuous missing data is represented by (β_m, φ_m) , in which β_m is the center angle and φ_m denotes the size of the angle range. In practice, only the angle ranges with large φ_m are considered to avoid noises. The size of the feature is expressed as

$$\|f_{\text{mis}}\| = \sum_m \varphi_m \quad (6)$$

Figure 3 shows the expression example of a missing data feature, as well as its appearance in scan data.

Figure 3. The expression of a missing data feature (left) and a feature in a scan scene (right).

For two different and adjacent robot poses, if the size of the missing data features f_{m1} and f_{m2} are big enough, there could be a quick way to estimate the rotation angle α , by minimizing the following matching error:

$$e = \underset{\alpha}{\operatorname{argmin}} \left(\frac{\|f_{m1} - \operatorname{rotate}(f_{m2}, \alpha)\|}{\|f_{m1}\| + \|f_{m2}\|} \right) \quad (7)$$

If the matching error e is smaller than a certain threshold, α could be regarded as the rough rotation angle between two poses. It is important to note that the missing data features should be big and stable enough for this processing. Fortunately, this is not uncommon indoors, especially when using a low-cost LRF sensor.

3. FFT-Based Scan-Matching

In this section, the scan-matching problem is transformed to an image registration problem, and it is solved by FFT.

3.1. Solution of One-Dimensional Fast Fourier Transform (1D FFT)

Considering two functions $f_1(x)$ and $f_2(x)$ with the following relationship

$$f_2(x) = f_1(x - x_0) \quad (8)$$

where x_0 is the shift between the two functions. Take the FFT of both sides of (8) and use related theorems to give

$$F_2(u) = F_1(u)e^{-j2\pi ux_0} \quad (9)$$

Taking the cross-power spectrum of (9):

$$\frac{F_1(u)F_2^*(u)}{|F_1(u)F_2^*(u)|} = e^{-j2\pi ux_0} \quad (10)$$

where F^* is the conjugation of F and $|\bullet|$ is magnitude. Taking the inverse FFT of both sides of (10) gives an impulse function. The translation parameter x_0 can be derived from the position of the magnitude of the impulse function [33].

Similarly, 2D FFT function relation can be written in this way:

$$f_2(x, y) = f_1(x - x_0, y - y_0) \quad (11)$$

where $f_1(x, y)$, $f_2(x, y)$ are two 2D functions and x_0, y_0 are displacements between them. We can also calculate the cross-power spectrum of (6):

$$\frac{F_1(u, v)F_2^*(u, v)}{|F_1(u, v)F_2^*(u, v)|} = e^{-j2\pi(ux_0 + vy_0)} \quad (12)$$

3.2. Rotation Parameters

Let $A(x, y)$ be the transformed scan image of $B(x, y)$, with rotation angle α , translation (x_0, y_0) , and scale factor λ :

$$A(x, y) = B[\lambda(x \cos \alpha + y \sin \alpha) - x_0, (-x \sin \alpha + y \cos \alpha) - y_0] \quad (13)$$

In this paper, there is no scaling relation between two scan images and $\lambda = 1$. We can write (13) as follows:

$$A(x, y) = B[(x \cos \alpha + y \sin \alpha) - x_0, (-x \sin \alpha + y \cos \alpha) - y_0] \quad (14)$$

Taking the FFT of both sides of (14) gives

$$|A(u, v)| = |B(u \cos \alpha + v \sin \alpha, -u \sin \alpha + v \cos \alpha)| \quad (15)$$

Performing polar coordinate transformation gives

$$u \cos \alpha + v \sin \alpha = \rho \cos(\theta - \alpha) \quad (16)$$

$$-u \sin \alpha + v \cos \alpha = \rho \sin(\theta - \alpha) \quad (17)$$

We can write (17) as follows:

$$|A(\rho \cos \theta, \rho \sin \theta)| = |B(\rho \cos(\theta - \alpha), \rho \sin(\theta - \alpha))| \quad (18)$$

Let

$$A(\rho, \theta) = |A(\rho \cos \theta, \rho \sin \theta)| \quad (19)$$

$$B(\rho, \theta) = |B(\rho \cos \theta, \rho \sin \theta)| \quad (20)$$

which gives

$$A(\rho, \theta) = B(\rho, \theta - \alpha) \quad (21)$$

Equation (21) is similar to the 1D FFT. Using the 1D FFT theorem, rotation angle α can be solved. In contrast to traditional image registration, there is no scaling relationship between scan images. Therefore, the rotation angle can be obtained by performing polar coordinate transformation without further conversion of the logarithmic coordinate.

3.3. Translation Parameters

After calculating the rotation parameter, the translation parameters can be solved by projection of the image in horizontal and vertical directions. Let $C(x, y)$ be the grid map after $B(x, y)$ is rotated by rotation angle α and its projection can be represented by:

$$C_x(y) = \int_{-\infty}^{\infty} C(x, y) dx \quad (22)$$

$$C_y(x) = \int_{-\infty}^{\infty} C(x, y) dy \quad (23)$$

Similarly, projection of $A(x, y)$ can be represented by

$$A_x(y) = \int_{-\infty}^{\infty} A(x, y) dx \quad (24)$$

$$A_y(x) = \int_{-\infty}^{\infty} A(x, y) dy \quad (25)$$

By taking a projection of the image and converting 2D to 1D, 1D FFT can be used to calculate the translation parameters.

From Equation (21) to (25), the rotation parameter and translation parameters are eventually solved by 1D FFT. The FFT scan-matching process is shown in Figure 4.

Figure 4. Fast Fourier Transform (FFT)-based scan-matching.

4. The FFT-ICP Scan-Matching Frame-Work

Although FFT-based scan-matching is robust to noises and does not need any initial pose, the computation cost and accuracy are difficult to satisfy at the same time. In this section, based on our previous work in [47] and [48], a new FFT-ICP-based scan-matching frame-work is introduced, which could be faster, more robust, and more appropriate for low-cost LRF sensors.

Firstly, after transferring the scan data into image data for further processing, the missing data features f_{m1} and f_{m2} are extracted. If the missing data features are obvious enough, for example, $\|f_{m1}\| > Th1$ and $\|f_{m2}\| > Th1$, where $Th1$ is a threshold indicating the total missing angle of a scan, the missing data features are matched at the first place to determine a rough rotation angle of the two scans (Section 2.C). If the matching error is small enough, the rotation angle estimated by this step could be an effective initial guess for further processing.

Secondly, if the FFT-based scan-matching method proposed in Section 3.3 has taken place for extracting the rotation and translation parameters (R_1, T_1) , it could be denoted that the step of the FFT-based method for estimating rotation angle could be bypassed, as the first step already has a good guess for rotation angle through the missing feature matching.

Then, we take the transformation (R_1, T_1) to the scene scan data and apply the ICP precise matching described in the following paragraph.

According to [26], ICP calculates the transformation through iterations. In each iteration step, the algorithm selects the closest points as correspondences and calculates the transformation for minimizing the equation.

$$\sum_{i=1}^n \|M_i - (RS_i + T)\|^2 \quad (26)$$

where R is the matrix composed of rotation parameters, T is the matrix composed of translation parameters, n is the number of corresponding points, M_i is the model point set and S_i is the scene point set.

Two factors need to be considered. One is the selection of the closest points, the other is the calculation of transformation in each ICP iteration. The time complexity of ICP is dominated by the time for selecting the closest points, and some enhancements have been proposed, such as the

commonly used K-d tree. The most used method that is available to calculate the transformation in each iteration is Singular Value Decomposition (SVD). We give a brief introduction of SVD in the following.

Introducing the centroids of the points to simplify (26):

$$\bar{M} = \frac{1}{n} \sum_{i=1}^n M_i, \bar{S} = \frac{1}{n} \sum_{i=1}^n S_i \quad (27)$$

$$M'_i = M_i - \bar{M}, S'_i = S_i - \bar{S} \quad (28)$$

We can write (26) as follows:

$$\begin{aligned} & \frac{1}{n} \sum_{i=1}^n \|M_i - RS_i\|^2 + \|T\|^2 - 2T^T \cdot (\bar{M} - R\bar{S}) \\ &= \frac{1}{n} \sum_{i=1}^n \|M_i - RS_i\|^2 - \|\bar{M} - R\bar{S}\|^2 + \|T - (\bar{M} - R\bar{S})\|^2 \\ &= \frac{1}{n} \sum_{i=1}^n \|S'_i\|^2 + \frac{1}{n} \sum_{i=1}^n \|M'_i\|^2 - \frac{2}{n} \sum_{i=1}^n (M'_i)^T \cdot (RS'_i) + \|T - (\bar{M} - R\bar{S})\|^2 \end{aligned} \quad (29)$$

In order to minimize the above, the algorithm has to maximize the third term and minimize the fourth term. The fourth term has its minimum for $T = \bar{M} - R\bar{S}$.

Writing the third term below:

$$\text{Trace} \left(\sum_{i=1}^n (RS'_i) (M'_i)^T \right) = \text{Trace}(RH) = \sum_{i=1}^n (M'_i)^T \cdot (S'_i) \quad (30)$$

The third term has its maximum for $R = VU^T$, where V and U are derived by the SVD $H = UDV^T$. From the above, the transformation matrices are

$$R = VU^T, T = \bar{M} - R\bar{S} \quad (31)$$

In each iteration, we estimate the transformation (R, T) , transform the scene point data using (R, T) , and calculate the matching error until the matching error is small enough or reaches the maximum iteration time. A total transformation estimation (R_{ICP}, T_{ICP}) is extracted by accumulating transformations during each iteration.

Finally, based on the rough estimation result (R_1, T_1) and precise estimation result (R_{ICP}, T_{ICP}) , the final result is calculated as follows:

$$\begin{cases} R = R_{ICP} \cdot R_1 \\ T = R_{ICP} \cdot T_1 + T_{ICP} \end{cases} \quad (32)$$

As introduced above, the FFT-ICP-based scan-matching frame-work is shown in Figure 5.

Figure 5. Proposed Fast Fourier Transform-Iterative Closest Point (FFT-ICP) scan-matching frame-work.

5. Experiment

5.1. Experimental Facilities and Settings

The experimental facilities are shown in Figure 6. The left is the Robot, the right is RPLidar-A1. RPLidar-A1 is a low-cost LRF sensor with 360° coverage field and range of up to six meters. The key parameters of RPLidar-A1 are listed in Table 1. The computer used was a laptop, with intel i3 CPU and 2 GB memory.

Table 1. The key parameters of RPLidar-A1.

Measuring distance	Ranging accuracy	Angle range	Angle resolution	Frequency	Price
0.15–6 m	0.001 m	0°–360°	1°	5.5 Hz	\$73

The experimental data were obtained by RPLidar-A1 under different indoor environments, including office, corridor, laboratory room, etc., as shown in Figure 7. The ground truth of real-world experiments was collected by manual measurement of key positions.

Different experiments were undertaken to evaluate the performance of matching accuracy, matching success rate, computational speed, and moving performance. The methods for comparison included NDT [37], ICP [26], original FFT [48], original FFT-ICP1 [47], and the FFT-ICP2 of this paper.

In the experiments, for FFT processing, the maximum range of the scan data for converting to an image was $d_{max} = 3.5$ m. The image size for original FFT [48] was set to 256×256 , while the image size for FFT-ICP1 [47] and FFT-ICP2 was set to 128×128 .

Figure 6. Experimental facilities, robot platform (left), RPLidar-A1 sensor (right).

Figure 7. Typical experimental environments: Office (a); Corridor (b); and Room (c).

5.2. Scan Matching

For scan matching, two different poses and their corresponding scan data were taken as one set. The ground truth of the poses were given by manual measurement. A series of such data sets of different environments and different conditions were used for the experiments.

Figure 8 shows an example of the scan-matching results of different steps with our proposed method. As can be seen from Figure 8b, the approach of estimating the rotation angle through missing data only performs well under certain scenes. After FFT processing, the rough poses were estimated, and they were close enough to the real pose as shown in Figure 8c. Finally, as can be seen from Figure 8d, the two scans were matched precisely.

Figure 8. An example of matching result with proposed method. (a) The input scan data; (b) rotation via the angle estimated by missing data features; (c) the matching result with rough posed estimate by FFT; and (d) the final result.

To evaluate the matching success rate under different conditions, the data sets were separated into 3 parts: small pose difference (average translation 0.3 m, rotation 10°), middle pose difference (average translation 0.5 m, rotation 20°), large pose difference (average translation 0.8 m, rotation 30°). The iteration times for matching was limited to 200. If the error of the final aligned poses were smaller than translation 0.1 m and rotation 3° , we judge them as successful matchings.

Another example of matching result via different methods is shown in Figure 9. For the input scans in Figure 9a, the method of ICP finally converged to a wrong result in Figure 9b, while NDT reported a matching fail in Figure 9c. In Figure 9d, the result of FFT is near to the truth, but it still has errors. FFT-ICP1 and FFT-ICP2 give approximate and accurate results here.

Figure 9. Example of matching results via different methods. (a) The input scans; (b) the result of ICP; (c) the result of Normal Distributions Transform (NDT); (d) the result of FFT; (e) the result of FFT-ICP1 (original); and (f) the result of FFT-ICP2 (current paper).

As shown in Figure 10, the success rate of the ICP method decreases very quickly when the difference of poses becomes larger, especially when the rotation angle becomes large. That is because ICP needs the scan points to overlap enough for matching. Although the NDT method performs more robustly than does ICP, it still cannot deal with the large difference of the poses. The FFT-based methods are more robust through large pose changes because they are theoretically unaffected by position changes. The matching fail for FFT-based methods mainly comes from the excessive change of total shape caused by different positions.

Figure 10. Results of the matching success rates for different methods.

Table 2 shows the average matching accuracy of different methods. Comparatively speaking, our original FFT method has larger errors than the other methods. This is mainly because its processing image is small. The FFT-ICP1 and FFT-ICP2 methods achieve better results because of their coarse-to-fine estimation.

Table 2. Results of the matching accuracy.

Method	Avg. error of x (m)	Variance of x error	Avg. error of y (m)	Variance of y error	Avg. Rotation error (deg)	Variance of rotation error
NDT [37]	0.041	0.027	0.043	0.022	0.78	0.32
ICP [26]	0.043	0.033	0.034	0.024	0.81	0.31
FFT [48]	0.052	0.058	0.047	0.045	1.02	0.67
FFT-ICP1 [47]	0.033	0.019	0.034	0.020	0.64	0.29
FFT-ICP2 [this paper]	0.040	0.021	0.037	0.023	0.75	0.28

5.3. Execution Efficiency

Further experiments were undertaken to evaluate the execution efficiency. The average run times for each step of the FFT-ICP2 method in this paper are shown in Table 3. In our work, in the step of estimating rotation angle, if the method of missing-data feature-matching in Section 2.3 (takes only 5 ms) has given a good result, then the processing of FFT for rotation estimation in Section 3.2 (takes 103 ms) can be bypassed. That saves a lot of time compared with FFT-ICP1.

Table 3. Average run-time for each step of FFT-ICP2.

Step	Obtain scan images	Estimate the rotation		Estimate the translations by FFT	Precise estimation by ICP
		By missing data features (Section 2.3)	By FFT for rotation (optional)		
Average Run Time	10 ms	5 ms	103 ms	82 ms	11 ms

The execution efficiency experiment was also undertaken for different methods under different environments. As shown in Table 4, both NDT and our proposed FFT approaches need much more time than the ICP method to achieve a more robust result. Meanwhile, the computational cost of FFT is inversely proportional to the number of grids and the computational accuracy. By down-sizing the image from 256×256 to 128×128 , the FFT-ICP1 has a faster in calculation speed than FFT. Meanwhile, for the places with obvious missing data features, such as corridors, the FFT-ICP2 proposed in this paper shows improvement over our previous methods.

Table 4. Average run-time for different methods.

Scenes	ICP (ms)	NDT (ms)	FFT (ms)	FFT-ICP1 (ms)	FFT-ICP2 (ms)
Room	23	108	355	198	201
Office	27	127	367	201	134
Corridor	26	120	365	197	101

5.4. Dynamic Localization

We also completed dynamic localization experiments based on the robot mobile platform. In the experiments, robot ran along a given trajectory under different moving speeds.

As shown in Figure 11, when the speed exceeds 0.15 m/s, the accuracy of the ICP algorithm is poor and the error is up to 0.5 m (tracking lost). In addition, due to the vibration of the robot, the curves generated by the ICP algorithm fluctuate greatly. In contrast, the algorithm proposed by this paper demonstrates robustness with speed; the location error is less than 0.2 m when the speed is

0.2 m/s. The curves generated by the proposed algorithm are smoother, so it also has a good robustness to vibration.

Figure 11. Localization results using ICP (a) and our FFT-ICP2 method (b) under different speeds.

Figure 12 shows the results of generating maps directly from a series of scans. In the experiments, robot ran at a linear speed up to 0.3 m/s and a rotation speed of $25^\circ/\text{s}$. The given robot trajectories are shown as red stars. As can be seen from Figure 12, the NDT method failed with rotation, while our proposed method gave a better result.

The experiments show that the proposed method in this paper leads to an improvement of robustness and speed, compared with our previous methods. This technique has proven to be helpful for constructing an on-line SLAM system.

Figure 12. The generated maps using NDT (a) and the proposed FFT-ICP2 method (b). The red stars represent the real locations of the robot.

6. Conclusions and Future Work

In this paper, a new approach of scan-matching and low-cost mobile robot localization is presented using a low-cost LRF sensor. The missing data caused by the out-of-measurement range are processed as features for estimating the rotation angle under certain typical scenes. FFT-based matching is applied to determine the rough pose. An effective implementation of an FFT-ICP-based method, in an attempt to reduce the computational cost, is proposed. The experiments show the proposed method has an advantage in computational cost over our previous methods. The proposed method is also more robust than traditional methods in long-distance matching, which suggests its potential for loop-closing detection in SLAM.

The drawback of this method is its detection range, because we only considered the obstacles near to the robot for matching. In wide scenarios, the current method may not work well. For the missing data, we only considered that they are caused by the out-of-measurement range. However, for low-cost LRF sensors, missing data can also be caused by on-total reflection or absorption of the laser. Scenes with a lot of glass are still a big challenge for all the laser-based localization and mapping approaches.

In the future, we will enhance the algorithm by scaling the scan image adaptively and developing quick loop-closing modules based on the image for the SLAM framework so that the algorithm can adopt to big scenes. We will also put forward a missing-data analysis to make full use of the information contained within the missing data.

Author Contributions: Conceptualization, W.X. and Y.O.; Data curation, L.Y.; Funding acquisition, Y.O.; Investigation, Y.O.; Methodology, G.J.; Software, G.L.; Supervision, Y.O.; Validation, L.Y.; Writing—original draft, G.J.

Funding: This work was supported by the National Natural Science Foundation of China (Grants No. U1613210), and the Shenzhen Fundamental Research Program (JCYJ20170413165528221).

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

References

1. Thrun, S.; Burgard, W.; Fox, D. *Probabilistic Robotics*; The MIT Press: Cambridge, UK, 2005.
2. Castellanos, J.A.; Martinez, J.M.; Neria, J.; Tardos, J.D. Simultaneous Map Building and Localization for Mobile Robots: A Multisensor Fusion Approach. In Proceedings of the 1998 IEEE International Conference on Robotics and Automation, Leuven, Belgium, 20 May 1998; Volume 2, pp. 1244–1249.
3. Zhao, Y.; Liu, F.; Wang, R. Location Technology of Indoor Robot Based on Laser Sensor. In Proceedings of the 2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 26–28 August 2016; pp. 683–686.
4. Desouza, G.N.; Kak, A.C. Vision for Mobile Robot Navigation: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2002**, *24*, 237–267. [[CrossRef](#)]
5. Se, S.; Lowe, D.; Little, J. Local and Global Localization for Mobile Robots using Visual Landmarks. In Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, Maui, USA, 29 October 2001; Volume 1, pp. 414–420.
6. Uchimoto, T.; Suzuki, S.; Matsubara, H. A Method to Estimate Robot's Location Using Vision Sensor for Various Type of Mobile Robots. In Proceedings of the 2009 International Conference on Advanced Robotics, Munich, Germany, 22–26 June 2009; pp. 1–6.
7. Mur-Artal, R.; Montiel, J.M.M.; Tardós, J.D. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [[CrossRef](#)]
8. Jan, S.; Gumhold, S.; Cremers, D. Real-Time Dense Geometry from a Handheld Camera. In *Dagm Conference on Pattern Recognition*; Springer: Berlin, Germany, 2010; pp. 11–20.
9. Engel, J.; Sturm, J.; Cremers, D. Semi-dense Visual Odometry for a Monocular Camera. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 1449–1456.
10. Kerl, C.; Sturm, J.; Cremers, D. Dense visual SLAM for RGB-D cameras. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 2100–2106.
11. Valiente, D.; Payá, L.; Jiménez, L.; Sebastián, J.; Reinoso, Ó. Visual Information Fusion through Bayesian Inference for Adaptive Probability-Oriented Feature Matching. *Sensors* **2018**, *18*, 2041. [[CrossRef](#)] [[PubMed](#)]
12. Valiente, D.; Gil, A.; Payá, L.; Sebastián, J.M.; Reinoso, Ó. Robust Visual Localization with Dynamic Uncertainty Management in Omnidirectional SLAM. *Appl. Sci.* **2017**, *7*, 1294. [[CrossRef](#)]
13. Nam, T.H.; Shim, J.H.; Cho, Y.I. A 2.5D Map-Based Mobile Robot Localization via Cooperation of Aerial and Ground Robots. *Sensors* **2017**, *17*, 2730. [[CrossRef](#)] [[PubMed](#)]
14. Nüchter, A. Parallelization of scan matching for robotics 3D mapping. In Proceedings of the 3rd European Conference on Mobile Robots, Freiburg, Germany, 19–21 September 2007.
15. Wolcott, R.W.; Eustice, R.M. Fast LIDAR localization using multiresolution Gaussian mixture maps. In Proceedings of the IEEE International Conference on Robotics and Automation, Seattle, WA, USA, 26–30 May 2015; pp. 2814–2821.

16. Grisetti, G.; Stachniss, C.; Burgard, W. Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters. *IEEE Trans. Robot.* **2007**, *23*, 34–46. [[CrossRef](#)]
17. Steux, B.; Hamzaoui, O.E. tinySLAM: A SLAM algorithm in less than 200 lines C-language program. In Proceedings of the International Conference on Control Automation Robotics & Vision, Venice, Italy, 23–25 November 2011; pp. 1975–1979.
18. Santos, J.M.; Portugal, D.; Rui, P.R. An evaluation of 2D SLAM techniques available in Robot Operating System. In Proceedings of the IEEE International Symposium on Safety, Security, and Rescue Robotics, Linköping, Sweden, 21–26 October 2013; pp. 1–6.
19. Konolige, K.; Grisetti, G.; Kümmerle, R.; Burgard, W.; Limketkai, B.; Vincent, R. Efficient Sparse Pose Adjustment for 2D mapping. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010; pp. 22–29.
20. Hess, W.; Kohler, D.; Rapp, H.; Andor, D. Real-time loop closure in 2D LIDAR SLAM. In Proceedings of the IEEE International Conference on Robotics and Automation, Stockholm, Sweden, 16–21 May 2016; pp. 1271–1278.
21. López, E.; García, S.; Barea, R.; Bergasa, L.M.; Molinos, E.J.; Arroyo, R.; Romera, E.; Pardo, S. A Multi-Sensorial Simultaneous Localization and Mapping (SLAM) System for Low-Cost Micro Aerial Vehicles in GPS-Denied Environments. *Sensors* **2017**, *17*, 802. [[CrossRef](#)] [[PubMed](#)]
22. Jung, J.; Kim, J.; Yoon, S.; Kim, S.; Cho, H.; Kim, C.; Heo, J. Bore-Sight Calibration of Multiple Laser Range Finders for Kinematic 3D Laser Scanning Systems. *Sensors* **2015**, *15*, 10292–10314. [[CrossRef](#)] [[PubMed](#)]
23. Papoutsidakis, M.; Kalovrektis, K.; Drosos, C.; Stamoulis, G. Design of an Autonomous Robotic Vehicle for Area Mapping and Remote Monitoring. *Int. J. Comput. Appl.* **2017**, *167*, 36–41. [[CrossRef](#)]
24. Ko, N.Y.; Kuc, T.Y. Fusing Range Measurements from Ultrasonic Beacons and a Laser Range Finder for Localization of a Mobile Robot. *Sensors* **2015**, *15*, 11050–11075. [[CrossRef](#)] [[PubMed](#)]
25. Oh, T.; Lee, D.; Kim, H.; Myung, H. Graph Structure-Based Simultaneous Localization and Mapping Using a Hybrid Method of 2D Laser Scan and Monocular Camera Image in Environments with Laser Scan Ambiguity. *Sensors* **2015**, *15*, 15830–15852. [[CrossRef](#)] [[PubMed](#)]
26. 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](#)]
27. Lu, F.; Milios, E. Robot pose estimation in unknown environments by matching 2d range scans. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 18–22 June 1994; pp. 935–938.
28. Bosse, M.C. ATLAS: A Framework for Large Scale Automated Mapping and Localization. Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA, USA, 2004.
29. Olson, E. Robust and Efficient Robotic Mapping. Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA, USA, 2008.
30. Lv, J.; Wang, Y.; Wu, K.; Dissanayake, G.; Kobayashi, Y.; Xiong, R. Planar scan matching using incident angle. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Vancouver, BC, Canada, 24–28 September 2017; pp. 4049–4056.
31. Diosi, A.; Kleeman, L. Fast laser scan matching using polar coordinates. *Int. J. Robot. Res.* **2007**, *26*, 1125–1153. [[CrossRef](#)]
32. Censi, A.; Iocchi, L.; Grisetti, G. Scan matching in the hough domain. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Barcelona, Spain, 18–22 April 2005.
33. Rfer, T. Using histogram correlation to create consistent laser scan maps. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Washington, DC, USA, 11–15 May 2002; pp. 625–630.
34. Bosse, M.; Roberts, J. Histogram matching and global initialization for laser-only slam in large unstructured environments. In Proceedings of the 2007 IEEE International Conference on Robotics and Automation, Roma, Italy, 10–14 April 2007; pp. 4820–4826.
35. Konolige, K.; Chou, K. Markov Localization using Correlation. In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 31 July–6 August 1999; Morgan Kaufmann Publishers Inc.: San Mateo, CA, USA, 1999; pp. 1154–1159.
36. Olson, E.B. Real-time correlative scan matching. In Proceedings of the IEEE International Conference on Robotics and Automation, Kobe, Japan, 12–17 May 2009; pp. 4387–4393.

37. Biber, P.; Strasser, W. The normal distributions transform: A new approach to laser scan matching. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 27–31 October 2003; Volume 3, pp. 2743–2748.
38. Lauer, M.; Lange, S.; Riedmiller, M. Calculating the perfect match: An efficient and accurate approach for robot self-localization. In *Robot Soccer World Cup*; Springer: Berlin/Heidelberg, Germany, 2005.
39. Sobreira, H.; Costa, C.M.; Sousa, I.; Rocha, L.; Lima, J.; Farias, P.C.M.A.; Costa, P.; Moreira, A.P. Map-Matching Algorithms for Robot Self-Localization: A Comparison Between Perfect Match, Iterative Closest Point and Normal Distributions Transform. *J. Intell. Robot. Syst.* **2018**, *92*, 1–14. [[CrossRef](#)]
40. Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
41. Bay, H.; Tuytelaars, T.; Van Gool, L. SURF: Speeded up robust features. *Comput. Vis. Image Understand. (CVIU)* **2008**, *110*, 346–359. [[CrossRef](#)]
42. Rublee, E.; Rabaud, V.; Konolige, K.; Bradski, G. ORB: An efficient alternative to SIFT or SURF. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; Volume 58, pp. 2564–2571.
43. Alliney, S.; Morandi, C. Digital Image Registration Using Projections. *IEEE Trans. Pattern Anal. Mach. Intell.* **1986**, *PAMI-8*, 222–233. [[CrossRef](#)]
44. Stone, H.S.; Orchard, M.T.; Chang, E.-C.; Martucci, S.A. A Fast Direct Fourier-based Algorithm for Subpixel Registration of Images. *IEEE Trans. Geosci. Remote Sens.* **2001**, *39*, 2235–2243. [[CrossRef](#)]
45. Gonzalez, R. Fourier Based Registration of Differentially Scaled Images. In Proceedings of the 2013 IEEE International Conference on Image Processing, Melbourne, Australia, 15–18 September 2013; pp. 1282–1285.
46. Guo, X.; Xu, Z.; Lu, Y.; Pang, Y. An Application of Fourier-Mellin Transform in Image Registration. In Proceedings of the Fifth International Conference on Computer and Information Technology (CIT'05), Shanghai, China, 21–23 September 2005; pp. 619–623.
47. Xi, W.; Ou, Y.; Feng, W.; Yu, G. An improved serial method for mobile robot SLAM. In Proceedings of the 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO), Macau, China, 5–8 December 2017; pp. 501–506.
48. Xi, W.; Ou, Y.; Peng, J.; Yu, G. A new method for indoor low-cost mobile robot SLAM. In Proceedings of the 2017 IEEE International Conference on Information and Automation (ICIA), Macau, China, 18–20 July 2017; pp. 1012–1017.

© 2018 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 (<http://creativecommons.org/licenses/by/4.0/>).