

Article

3D Environment Mapping with a Variable Resolution NDT Method

Yang Feng , Zhiyuan Gao, Jinghan Zhang, Hang Shi and Yangmin Xie *

School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200444, China

* Correspondence: xieym@shu.edu.cn; Tel.: +86-182-2152-2016

Abstract: With the continuous development of the 3D LiDAR (Light Detection And Ranging) mapping algorithm and its application in various fields, the size of the point cloud map becomes a bottleneck that limits the 3D LiDAR mapping algorithm from running for a long time. In this paper, a 3D LiDAR mapping method based on scan-to-map and variable resolution NDT (normal-distributions transform) registration is proposed. When updating the global map, variable resolution processing can reduce the size of the global map and improve the accuracy of map construction. In addition, the size of the map created by the mapping algorithm is proportional to the size of the space and does not grow infinitely over time. The mapping experiments using a rotating LiDAR in the room, corridor, and outdoor environments show that the algorithm has higher mapping accuracy and smaller map size than without considering the variable resolution strategy. The experimental results of the map construction for a long time in an appropriate test area illustrate that the map built by the algorithm does not grow infinitely with time in the fixed space. In summary, by adjusting the map resolution adaptively according to the curvature of different areas in the 3D LiDAR mapping process, the proposed variable resolution strategy can maintain the size of the global map almost proportional to the size of the space. Moreover, the mapping accuracy can be improved as well.

Keywords: 3D LiDAR mapping; scan-to-map; NDT; variable resolution



Citation: Feng, Y.; Gao, Z.; Zhang, J.; Shi, H.; Xie, Y. 3D Environment Mapping with a Variable Resolution NDT Method. *Machines* **2022**, *10*, 1200. <https://doi.org/10.3390/machines10121200>

Academic Editor: Antonios Gasteratos

Received: 14 November 2022

Accepted: 6 December 2022

Published: 10 December 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 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

3D environment map construction technology has a wide range of application prospects, such as precision agriculture [1], building reconstruction [2], search and rescue [3], tree surveying [4], terrain mapping [5], historical building appearance mapping [6], and so on. Therefore, it is of great significance to study the 3D environment map construction technology.

LiDAR (Light Detection And Ranging) and camera are mainly used for 3D mapping. For example, a monocular camera is used for real-time UAV (Unmanned Aerial Vehicle) dense mapping [7], an RGB-D camera is used for indoor unbounded RGB-D dense point cloud 3D reconstruction with high accuracy [8], and a LiDAR is used for environment map construction to cope with challenging agricultural scenarios [9]. The camera can only be used in a well-lit environment, while the use of LiDAR is almost unaffected by the environment. Compared with the camera, LiDAR has higher accuracy and a larger field of view, so we use 3D LiDAR to conduct research on 3D map construction.

3D environment maps are represented in many different ways, including point cloud [10], mesh [11], surfel [4], grid [12,13], polarized LiDAR Map [14], etc. The point cloud is an effective way to represent the real environment. With the rapid development of LiDAR, there is more and more research on LiDAR-based mapping algorithms, such as [15–17]; all are based on research of 3D LiDAR mapping algorithms. Since the point cloud can represent the surrounding environment simply and intuitively, it is more convenient to use than other types of map representation, so we use point cloud as the map representation.

When using the point cloud as the map representation, the map will become denser and denser over time, even with the limited environment. The conventional methods to

solve this problem include removing redundant points according to the distance between point clouds [18,19] or the time interval over the capturing time of the point cloud [20]. Some researchers also tackle the above question by detecting the overlap degree of sub-maps and carrying out map fusion [21–23]. The method of removing redundant points in the map according to the distance between point clouds is similar to uniform downsampling, which is convenient to implement, but it cannot distinguish objects with different features. The strategy of removing redundant points according to the time when the point cloud is added to the global map does not make use of the previous point cloud information, which is not suitable for constructing a point cloud map. The method of detecting the overlap degree of the sub-maps requires maintaining the sub-maps, which is difficult to implement.

To solve the problem of significant redundancy caused by repeated scanning of the same space, we propose to process the map with a variable resolution strategy and adaptively adjust the point cloud density of different locations in the map according to their curvature. Thus, different locations in the map have reasonable point cloud density. This ensures that the map size varies with the size of the space rather than growing infinitely over time. In addition, in the process of variable resolution processing of the map, redundant points are removed according to the probability density value of each point in the global map, which is helpful in improving the accuracy of map construction.

The main contributions of this paper are as follows:

- A novel strategy of variable resolution processing for point cloud maps is proposed, which can reduce the map size effectively without decreasing the mapping accuracy;
- The size of the map generated with the proposed strategy increases proportionately to the environmental extent rather than the scanning time;
- The experimental results in both structured and unstructured environments demonstrated the efficiency of the proposed method.

The rest of the paper is organized as follows. Section 2 reviews some research related to this paper. Section 3 introduces the algorithm of the whole system. Section 4 introduces the experiment results and discussion. The experimental environment includes a room environment, a corridor environment, an outdoor environment, and a conference room environment. Section 5 summarizes the paper and puts forward suggestions for future modification.

2. State of the Art

This section introduces some 3D mapping systems and measures to reduce the redundant points in the map. The strategies for removing redundant points in the map include removing redundant points according to the distance between point clouds or the time when the point cloud was added to the global map, detecting the overlap degree of sub-maps and merging the sub-maps with higher overlap degree, point cloud compression, and multi-resolution maps, etc.

Removing redundant points according to the distance between point clouds is a common method to remove redundant points. There are different ways to implement it. In paper [18], before adding a new point to the global map, it is necessary to search within a certain radius of the point, and this point will be added to the map only if no results are returned. If there are already points within the radius, only the point with the least uncertainty is kept. In paper [19], the points whose distance is less than a certain threshold are directly removed. The nature of this way of removing redundant points is similar to the uniform downsampling of the point cloud, but the disadvantage is that different features cannot be distinguished.

The strategy of removing redundant points based on when the point cloud was added to the map is crude. In paper [20], a planar feature is stored in each voxel of the global map. When the uncertainty of the planar feature in a voxel converges, all the historical points are discarded, and the estimated planar parameters are retained. Once a new point appears, only the latest 10 points are kept, and the new plane features composed of these 10 points

are calculated. This way of updating the map only uses the newly obtained information of a few point clouds and does not consider the information of the previous point clouds at the location. In this case, the global map is represented by plane elements, and the strategy of removing redundant points according to the time when the point cloud was added to the global map is feasible, but it may not be appropriate for the point cloud map.

It is a good way to reduce the size of the map by merging sub-maps. The map update strategy in paper [21] is to search all the sub-maps involved in the closed loop when a closed loop is detected, and then fuse these sub-maps together. Sub-map fusion can save memory by preventing new sub-maps from being created when the same space is revisited. This has the advantage of making the reconstruction complexity proportional to the amount of space explored, rather than the duration of the exploration. However, this scheme only works when the closed loop is detected, and searching all the sub-maps after the closed loop is detected also needs more time. Paper [22] introduces a sub-map fusion strategy based on overlapping scan space on the basis of paper [21]. Sub-map overlap is calculated by comparing volume reconstruction and occupancy information stored in each sub-map. If the overlap of two sub-maps exceeds the threshold, it indicates that there is significant redundancy between the sub-maps, and map fusion is required. The calculation of sub-map overlap also requires traversing a large number of sub-maps, resulting in a large amount of computation. In paper [23], a method to calculate the map increment is proposed, and only the incremental part is added when the map is updated. It is also a time-consuming operation to calculate the map increment in this method.

There have also been some studies on reducing the size of the map by compressing the point cloud. In paper [24], a Recurrent Neural Network is used to compress the point cloud. In paper [25], a clustering method is used to compress the point cloud. In paper [26], a voxelized point cloud compression scheme is proposed. In paper [27], a point cloud coding scheme is proposed. Although these point cloud compression schemes can significantly reduce the size of point cloud maps, they are more suitable for point cloud storage and transmission than for online map construction, because point cloud maps need to be compressed and decompressed frequently when this method is used for online mapping.

In some studies, multi-resolution maps are used to remove redundant information in maps. In paper [28], a multi-resolution voxel map representation method is proposed to observe the occupancy probability of different voxels, and only the voxels with higher occupancy probability are retained. This method is suitable for occupancy voxel maps. In paper [29], different resolutions are set according to the distance between the point cloud and the sensor. This method is suitable for scenarios where the sensor is fixed, but it is not easy to implement when the sensor is moving.

The variable resolution strategy proposed in this paper can consider different types of features in the map, and adopt different resolutions for different positions adaptively, which can better express the information in the environment. In addition, the map update is carried out in the local area covered by the point cloud of the current LiDAR scan, which is more convenient to implement. These features can be very helpful for some applications, such as building reconstruction and search rescue. The proposed mapping algorithm can significantly reduce the map data size during building reconstruction and search rescue while preserving adequate environmental information.

3. Method

This section introduces the entire 3D LiDAR mapping system in detail. Section 3.1 describes the overall structure of the algorithm process, Section 3.2 describes the scan-to-map based NDT (normal-distributions transform) registration algorithm, and Section 3.3 describes the variable resolution update strategy of the global map.

For clarity, some denotations used in this paper are presented as follows. The coordinate system $\{M\}$ is defined as the global coordinate system, and the point cloud in $\{M\}$ is denoted as ${}^M C$, where ${}^M C = \{{}^M \mathbf{p}_0, {}^M \mathbf{p}_1, {}^M \mathbf{p}_2, \dots\}$, ${}^M \mathbf{p}_i \in \mathbb{R}^3$ is a point in point cloud ${}^M C$. $\{L_k\}$ is the LiDAR frame at time k , the point cloud in frame $\{L_k\}$ is denoted as ${}^{L_k} C$, where

$L_k C = \{L_k \mathbf{p}_0, L_k \mathbf{p}_1, L_k \mathbf{p}_2, \dots\}$, $L_k \mathbf{p}_i \in \mathbb{R}^3$ is a point in point cloud $L_k C$. The 3D geometry transformation from frame $\{L_k\}$ to global frame $\{M\}$ is denoted as ${}^M \mathbf{T}_{L_k} \in SE(3)$, where $SE(3)$ is the special Euclidean group.

3.1. System Overview

The pipeline of the 3D LiDAR mapping system proposed in this paper is shown in Figure 1. The measurement error of LiDAR scanning is relatively large at a far distance, and the measured points with a small range are induced by the vehicle body itself. Therefore, a distance filter is applied to remove the aforementioned disturbing measurements. The filtered measurement is $L_k C$. Then, the constant turn rate and velocity (CTRV) model [30] is used to correct the distortion in $L_k C$, which results in a new point cloud $L_k C'$. Then, the reference point cloud ${}^M C_{ref}$ is extracted from the global map ${}^M C$ to overlap with $L_k C'$. A scan-to-map NDT algorithm is used to register $L_k C'$ and ${}^M C_{ref}$, and the transformation ${}^M \mathbf{T}_{L_k}$ from frame $\{L_k\}$ to the global frame $\{M\}$ is obtained. $L_k C'$ can be fine-tuned using the registration results, which outputs a corrected point cloud $L_k C''$. $L_k C''$ is merged to ${}^M C$ with an adaptive variable resolution strategy, which, as it is mentioned above, provides simplified and accurate environment models.

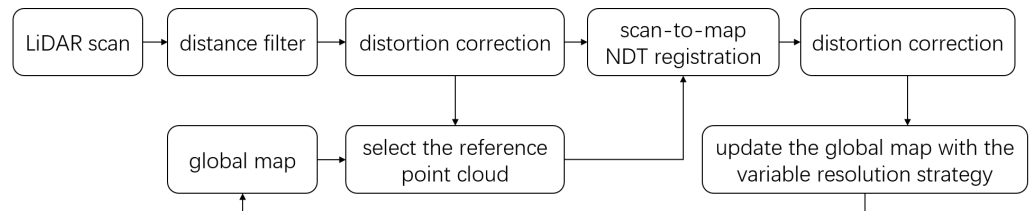


Figure 1. The pipeline of the 3D LiDAR mapping system.

The global map is managed using an Octree [31] for efficient voxel queries, where the point cloud belonging to a voxel is stored in the corresponding cell. The voxel-related properties, such as the mean and covariance of the local point cloud, the curvature, and the number of the points, are stored in a hash table [32], denoted as H , for fast information extraction.

3.2. Scan-to-Map NDT Registration

Compared with the scan-to-scan registration method, the scan-to-map registration method can largely avoid cumulative registration errors [33], so the scan-to-map method is used for point cloud registration in this paper.

3.2.1. Select the Global Reference Cloud

To reduce the computational burden, the scan-to-map registration method in this paper is not to register the current point cloud $L_k C'$ directly with the global point cloud ${}^M C$ but to select a sub-map point cloud ${}^M C_{ref}$ in ${}^M C$ that has overlapping areas with $L_k C'$.

$L_k C'$ is converted to the global frame $\{M\}$ with Equation (1), and denoted by ${}^M C'$.

$${}^M \mathbf{p}' = {}^M \mathbf{R}_{L_k} \cdot L_k \mathbf{p}' + {}^M \mathbf{t}_{pre} \quad (1)$$

where $L_k \mathbf{p}' \in L_k C'$, ${}^M \mathbf{p}' \in {}^M C'$. ${}^M \mathbf{p}'$ are the points after $L_k \mathbf{p}'$ is transformed to frame $\{M\}$, ${}^M \mathbf{R}_{L_k} \in \mathbb{R}^{3 \times 3}$ and ${}^M \mathbf{t}_{pre} \in \mathbb{R}^3$ are the rotation matrix and translation vector of the predicted frame $\{L_k\}$ with respect to the global frame $\{M\}$. ${}^M \mathbf{R}_{L_k}$ and ${}^M \mathbf{t}_{pre}$ are calculated from Equations (2) and (3), respectively, using the CTRV model.

$${}^M \mathbf{R}_{L_k} = {}^M \mathbf{R}_{L_{k-1}} \cdot L_{k-2} \mathbf{R}_{L_{k-1}} \quad (2)$$

$${}^M \mathbf{t}_{pre} = {}^M \mathbf{R}_{L_{k-1}} \cdot L_{k-2} \mathbf{t} + {}^M \mathbf{t}_{L_{k-1}} \quad (3)$$

where ${}^M_{L_{k-1}}\mathbf{R} \in \mathbb{R}^{3 \times 3}$ and ${}^M_{L_{k-1}}\mathbf{t} \in \mathbb{R}^3$ are the rotation matrix and translation vector of the frame $\{L_{k-1}\}$ with respect to the frame $\{M\}$, and ${}^{L_{k-2}}_{L_{k-1}}\mathbf{R} \in \mathbb{R}^{3 \times 3}$ and ${}^{L_{k-2}}_{L_{k-1}}\mathbf{t} \in \mathbb{R}^3$ are the rotation matrix and translation vector of the frame $\{L_{k-1}\}$ with respect to the frame $\{L_{k-2}\}$.

In order to guarantee a sufficient overlap between ${}^MC'$ and ${}^MC_{ref}$ in the subsequent registration process, 26 voxels V_{adj} adjacent to V_i in the global map point cloud MC are searched and added to the Octree of ${}^MC_{ref}$. Since the pose discrepancy between ${}^MC'$ and MC is limited after applying the CTRV model, a single layer expansion with 26 adjacent voxels is enough to ensure an overlap between ${}^MC'$ and ${}^MC_{ref}$. Only the mean and covariance matrix of the point clouds in voxels are useful in the NDT registration process. Therefore, we only need to record the voxel index in the Octree of ${}^MC_{ref}$, and the corresponding voxel properties, including the mean and covariance matrix, can be quickly found in the global hash table H.

3.2.2. The NDT Registration

The NDT is a method of compactly representing a surface. The transform maps a point cloud to a smooth surface representation, described as a set of local probability density functions (PDFs), each of which describes the shape of a section of the surface. The NDT registration is to find ${}^M_{L_k}\mathbf{T}$ that maximizes the likelihood that the points of the current scan lie on the reference scan surface [34].

The initial guess of the ${}^M_{L_k}\mathbf{T}$ is denoted as ${}^M_{L_k}\mathbf{T}_{pre}$, which includes ${}^M_{L_k}\mathbf{R}_{pre}$ and ${}^M_{L_k}\mathbf{t}_{pre}$. Denote the difference between the optimal solution of ${}^M_{L_k}\mathbf{T}$ and the ${}^M_{L_k}\mathbf{T}_{pre}$ as $\Delta\mathbf{T}$, $\Delta\mathbf{T} \in SE(3)$. The likelihood function is defined as

$$\Psi = \prod_{i=1}^n \zeta(h(\Delta\mathbf{T}, {}^M\mathbf{p}'_i)) \quad (4)$$

or, equivalently, in its negative logistic form:

$$-\log\Psi = -\sum_{i=1}^n \log(\zeta(h(\Delta\mathbf{T}, {}^M\mathbf{p}'_i))) \quad (5)$$

where ${}^M\mathbf{p}'_i \in {}^MC'$. The definition of $h(\Delta\mathbf{T}, {}^M\mathbf{p}'_i)$ is given in Equation (6), and the definition of $\zeta(\mathbf{p})$ is given in Equation (7).

$$h(\Delta\mathbf{T}, {}^M\mathbf{p}'_i) = \Delta\mathbf{T} \cdot {}^M\mathbf{p}'_i \quad (6)$$

$$\zeta(\mathbf{p}) = \frac{1}{(2\pi)^{3/2} |\Sigma_i|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{p} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\mathbf{p} - \boldsymbol{\mu}_i)\right\} \quad (7)$$

where $\zeta(\mathbf{p})$ is the probability density function corresponding to voxel V_i of point cloud ${}^MC_{ref}$, $\mathbf{p} \in \mathbb{R}^3$. $\boldsymbol{\mu}_i \in \mathbb{R}^3$ is the mean of points in voxel V_i , and $\Sigma_i \in \mathbb{R}^{3 \times 3}$ is the covariance matrix of points in voxel V_i .

Points with small probability density significantly deteriorate the registration results. A mixture of a normal distribution and a uniform distribution can be used to approximate Equation (5) to eliminate the influence of outliers on the results.

$$f(\Delta\mathbf{T}) = -\sum_{i=1}^n g(h(\Delta\mathbf{T}, {}^M\mathbf{p}'_i)) \quad (8)$$

where the definition of $g(\mathbf{p})$ is given in Equation (9).

$$g(\mathbf{p}) = -d_1 \cdot \exp\left(-\frac{d_2}{2}(\mathbf{p} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\mathbf{p} - \boldsymbol{\mu}_i)\right) \quad (9)$$

where d_1 and d_2 are constants. The readers are referred to the article [34] for details.

The Newton method is used to solve the following nonlinear optimization problems.

$$\Delta \mathbf{T}^* = \underset{\Delta \mathbf{T}}{\operatorname{argmin}} f(\Delta \mathbf{T}) \tag{10}$$

$$\Delta \mathbf{T}^* = \begin{bmatrix} \Delta \mathbf{R}^* & \Delta \mathbf{t}^* \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \tag{11}$$

where $\Delta \mathbf{R}^* \in \mathbb{R}^{3 \times 3}$ is the rotation matrix and $\Delta \mathbf{t}^* \in \mathbb{R}^3$ is the translation vector.

Accordingly, the rotation ${}^M_L \mathbf{R}$ and translation ${}^M_L \mathbf{t}$ of the LiDAR frame can be updated, as shown in Equations (12) and (13).

$${}^M_L \mathbf{R} = \Delta \mathbf{R}^* \cdot {}^M_{L_k} \mathbf{R}_{pre} \tag{12}$$

$${}^M_L \mathbf{t} = \Delta \mathbf{R}^* \cdot {}^M_{L_k} \mathbf{t}_{pre} + \Delta \mathbf{t}^* \tag{13}$$

3.3. Map Maintenance

With the pose of the current scan registered, the new point cloud needs to be merged into the global map, which is called global map updating. The first step is to refine the point cloud correction for $L_k C$ with the optimized registration results. The rotation matrix ${}^{L_{k-1}}_{L_k} \mathbf{R}$ and translation vector ${}^{L_{k-1}}_{L_k} \mathbf{t}$ between the frame k and $k - 1$ is calculated in Equations (14) and (15).

$${}^{L_{k-1}}_{L_k} \mathbf{R} = {}^M_{L_{k-1}} \mathbf{R}^{-1} \cdot {}^M_{L_k} \mathbf{R} \tag{14}$$

$${}^{L_{k-1}}_{L_k} \mathbf{t} = {}^M_{L_{k-1}} \mathbf{R}^{-1} \cdot ({}^M_{L_k} \mathbf{t} - {}^M_{L_{k-1}} \mathbf{t}) \tag{15}$$

The re-corrected scan $L_k C''$ is converted to the global frame $\{M\}$ with Equation (16), and denoted as ${}^M C''$.

$${}^M \mathbf{p}'' = {}^M_{L_k} \mathbf{R} \cdot {}^{L_k} \mathbf{p}'' + {}^M_{L_k} \mathbf{t} \tag{16}$$

where ${}^{L_k} \mathbf{p}'' \in L_k C''$, ${}^M \mathbf{p}'' \in {}^M C''$.

The essence of our variable resolution method is to maintain a geometric-adaptive point density in the map updating process. Denote the changed voxel set as V . Firstly, the curvature α_i of the point cloud in voxel $V_i \in V$ is calculated according to Equation (17).

$$\alpha_i = \frac{\lambda_{0_i}}{\lambda_{0_i} + \lambda_{1_i} + \lambda_{2_i}} \tag{17}$$

where λ_{0_i} , λ_{1_i} and λ_{2_i} are the eigenvalues of the covariance matrix of the point cloud in voxel V_i , and satisfy $\lambda_{0_i} \leq \lambda_{1_i} \leq \lambda_{2_i}$.

Then, the density upper limit ρ_{lim_i} of the point cloud in voxel V_i is calculated according to Equation (18).

$$\rho_{lim_i} = f_{max_density}(\alpha_i) = \begin{cases} \rho_{min} & \alpha_i < \alpha_{min} \\ \eta \cdot \alpha_i & \alpha_{min} \leq \alpha_i \leq \alpha_{max} \\ \rho_{max} & \alpha_i > \alpha_{max} \end{cases} \tag{18}$$

where ρ_{min} is the lower bound of ρ_{lim_i} , ρ_{max} is the upper bound of ρ_{lim_i} , η is a user-defined coefficient, α_{min} is the lower bound of α_i , and α_{max} is the upper bound of α_i . They can be tuned under the constraints $\alpha_{min} = \frac{\rho_{min}}{\eta}$ and $\alpha_{max} = \frac{\rho_{max}}{\eta}$. The higher the curvature of the point cloud in the voxel, the higher the upper limit of the density of the point cloud. If we need a denser map, we can set ρ_{min} , ρ_{max} , and η larger, and vice versa.

Then, the upper limit n_{max_i} of points number in voxel V_i can be calculated, as shown in Equation (19).

$$n_{max_i} = \rho_{lim_i} \cdot L^3 \tag{19}$$

where L is the edge length of the voxel.

If the point number n_i after map merging is larger than n_{max_i} , it implies that redundant points exist in the V_i and need to be removed. Firstly, $\gamma * n_{max_i}$ points with larger probability density values are separated as the maintaining points, then $(1 - \gamma) * n_{max_i}$ points are randomly selected from the left part and treated as the maintaining points as well, the other points which do not belong to the maintaining points will be removed. Here, γ is an adjustable coefficient. The probability density value of each point in voxel V_i is calculated as shown in Equation (7). In the process of NDT registration, outliers will have a certain impact on the registration results, so reserving the points with high probability density will help improve the accuracy of NDT registration and then improve the accuracy of map construction.

The map maintenance process described above is concluded as in Algorithm 1. In addition, we need to update the mean μ_i and covariance matrix Σ_i of the new point cloud in each voxel $V_i \in V$ in the global hash table H for further registration usage.

Algorithm 1: Variable Resolution Strategy

Data: voxel set V in global map, voxel edge length L , parameter γ
Result: voxel set V after variable resolution processing.
for voxel $V_i \in V$ **do**
 query point number n_i of V_i ;
 compute curvature α_i of V_i ;
 compute max point cloud density ρ_{lim_i} ;
 $n_{max_i} = \rho_{lim_i} * L^3$;
 if $n_i \leq n_{max_i}$ **then**
 | continue;
 else
 | separate $\gamma * n_{max_i}$ points as maintaining points whose probability density values are larger;
 | select $(1 - \gamma) * n_{max_i}$ points in the left points randomly as the maintaining points as well;
 | remove the remaining points that do not belong to the maintaining points;
 end
end
Return V

4. Experiment Results and Discussion

We verified the variable resolution NDT method in three different environments (room environment, corridor environment, and outdoor environment) for performance evaluation. In addition, we carried out a long-time experiment to illustrate the advantages of our mapping algorithm in decreasing the map data size.

4.1. Equipment Introduction

In the experiment, a mobile robot equipped with a 16-line rotating LiDAR (as shown in Figure 2a) was used to conduct the environment mapping experiment. The LiDAR can rotate around the X-axis shown in Figure 2a with respect to the vehicle body to obtain a larger field of view for environmental mapping. Specific device parameters are shown in Table 1. During the data-capturing process, the robot navigates along the given path, and the LiDAR captures the point cloud continuously. The ground truth of the environment is captured by a high-accuracy laser scanner system that can rotate around its Z-axis, as shown in Figure 2b, whose system accuracy is 0.003 m in the range of 5 m. Specific device parameters are shown in Table 2.

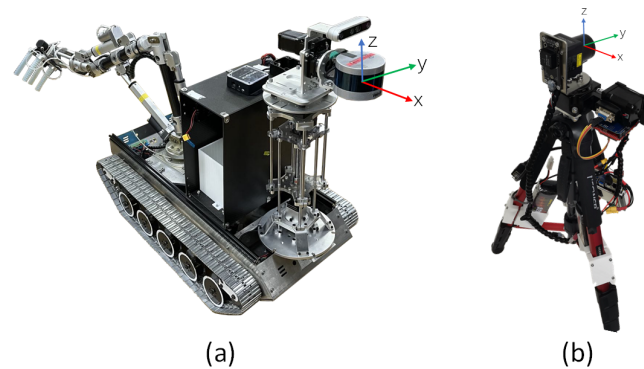


Figure 2. The equipment used in the experiment. (a) shows the mobile robot equipped with a 16-line rotating LiDAR. (b) shows the portable laser scanner used in the experiment to obtain the environmental truth value.

Table 1. The parameters of the spin LiDAR equipped on the mobile robot.

Sensor Type	Line Number	FOV	Scan Frequency	Frequency around X Axis	System Accuracy
VLP-16	16	270°	10 Hz	0.2 Hz	0.006 m *

* System accuracy in the range of 5 m.

Table 2. The parameters of the portable laser scanner.

Sensor Type	Line Number	FOV	Scan Frequency	Rotate Range around Z Axis	System Accuracy
Hokuyo UST-30LX	1	270°	40 Hz	180°	0.003 m *

* System accuracy in the range of 5 m.

4.2. Mapping Accuracy Analysis

To verify the efficiency of the variable resolution strategy on the accuracy of mapping, we carried out experiments in the room, corridor, and outdoor environments, respectively, and collected the environment truth value to compare the point-to-point map error with the algorithm without variable resolution strategy.

4.2.1. Room Environment

The room environment mapping experiment is carried out in the scene as shown in Figure 3. The scene is a laboratory with some experimental equipment and desks placed against the wall.

The environment ground truth is shown in Figure 4(a-1,a-2), the mapping result without considering the variable resolution strategy is shown in Figure 4(b-1,b-2), and the mapping result considering the variable strategy is shown in Figure 4(c-1,c-2). The full map is on the left and the interior details are on the right. Because the LiDAR can scan through the glass to get the outdoor scene, the map is not only the indoor scene but also part of the outdoor railing. There is no point cloud on the ground in the upper right part of Figure 4(b-2,c-2), because the mobile robot moves in the lower left part of the scene when collecting the point cloud data. It can be seen from Figure 4(c-1,c-2) that the density of the point cloud in different parts of the map is different. The density of the point cloud is low in areas with small curvature (such as ceiling, wall, and floor) and high in areas with large curvature (such as the experimental equipment). This effectively reduces the number of redundant points in the map.



Figure 3. The scene of the room environment.

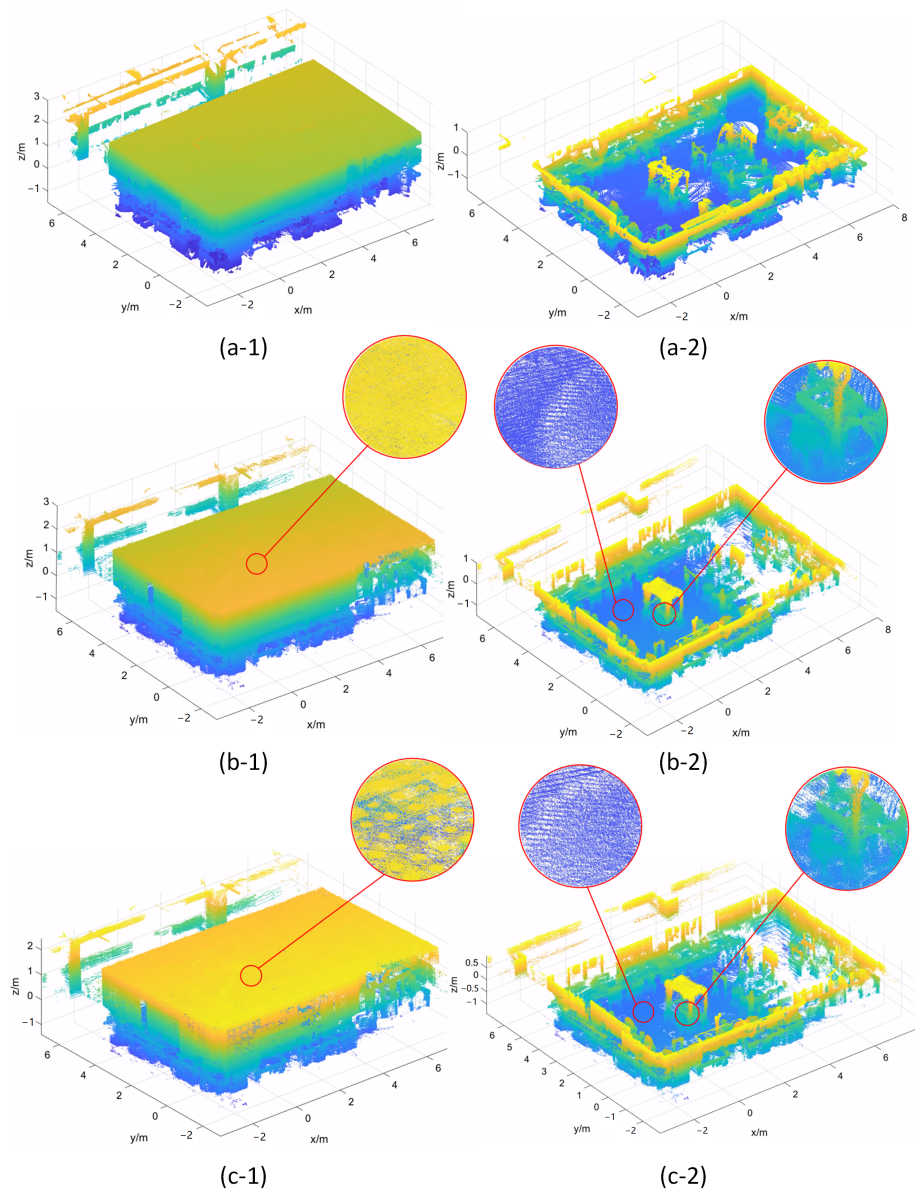


Figure 4. The result of the room environment mapping experiment (colored by depth). (a-1,a-2) show the ground truth. (b-1,b-2) show the mapping result without variable resolution strategy. (c-1,c-2) show the mapping result with variable resolution strategy. The full maps are on the left, and the interior details are on the right.

The CloudCompare software is used to align the point clouds shown in Figure 4(b-1,c-1) with the ground truth shown in Figure 4(a-1), and then, we randomly select some areas shown in Figure 5 to calculate the mean absolute error (MAE). The results are shown in Table 3. We selected a total of 8 areas to calculate the mapping error, among which areas 1–3 are the ceiling, areas 5 and 8 are the walls, area 4 is a piece of experimental equipment, and areas 6 and 7 are the desks. The mapping errors in these areas are smaller when the variable resolution processing strategy is used than when the strategy is not used. The accuracy of area 2 is improved most obviously, and the error is reduced by 14.58%.

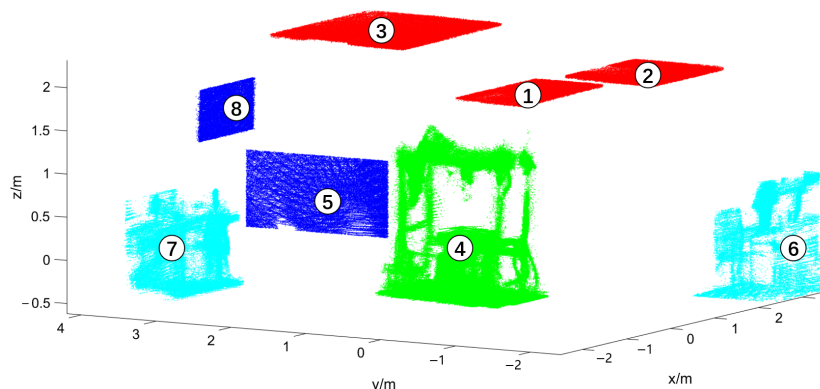


Figure 5. The selected areas of the room environment used to calculate the mapping error. Several kinds of regions are selected and colored separately, with each type of region colored in the same color. Areas 1–3 are the ceiling, areas 5 and 8 are the walls, area 4 is a piece of experimental equipment, and areas 6 and 7 are the desks.

Table 3. The mapping error (MAE/m) comparison of the room environment.

Area Index	Without Variable Resolution	With Variable Resolution	Area Index	Without Variable Resolution	With Variable Resolution
1	0.0039	0.0031	5	0.0055	0.0052
2	0.0048	0.0041	6	0.0115	0.0112
3	0.0062	0.0061	7	0.0097	0.0093
4	0.0064	0.0063	8	0.0046	0.0042

Smaller errors are in bold.

4.2.2. Corridor Environment

The corridor environment mapping experiment is carried out in the scene as shown in Figure 6. The scene is a U-shaped corridor with walls on one side and railings on the other.

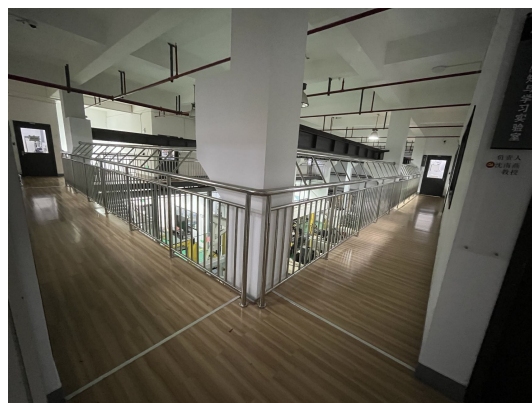


Figure 6. The scene of the corridor environment.

The environment ground truth is shown in Figure 7(a-1,a-2), the mapping result without considering the variable resolution strategy is shown in Figure 7(b-1,b-2), and the mapping result considering the variable strategy is shown in Figure 7(c-1,c-2). It can be seen that after considering the variable resolution strategy for the map, the density of the point cloud in the areas with low curvature (such as ceiling and floor) in the map is low, and the density of the point cloud in the areas with high curvature (such as railing) is high.

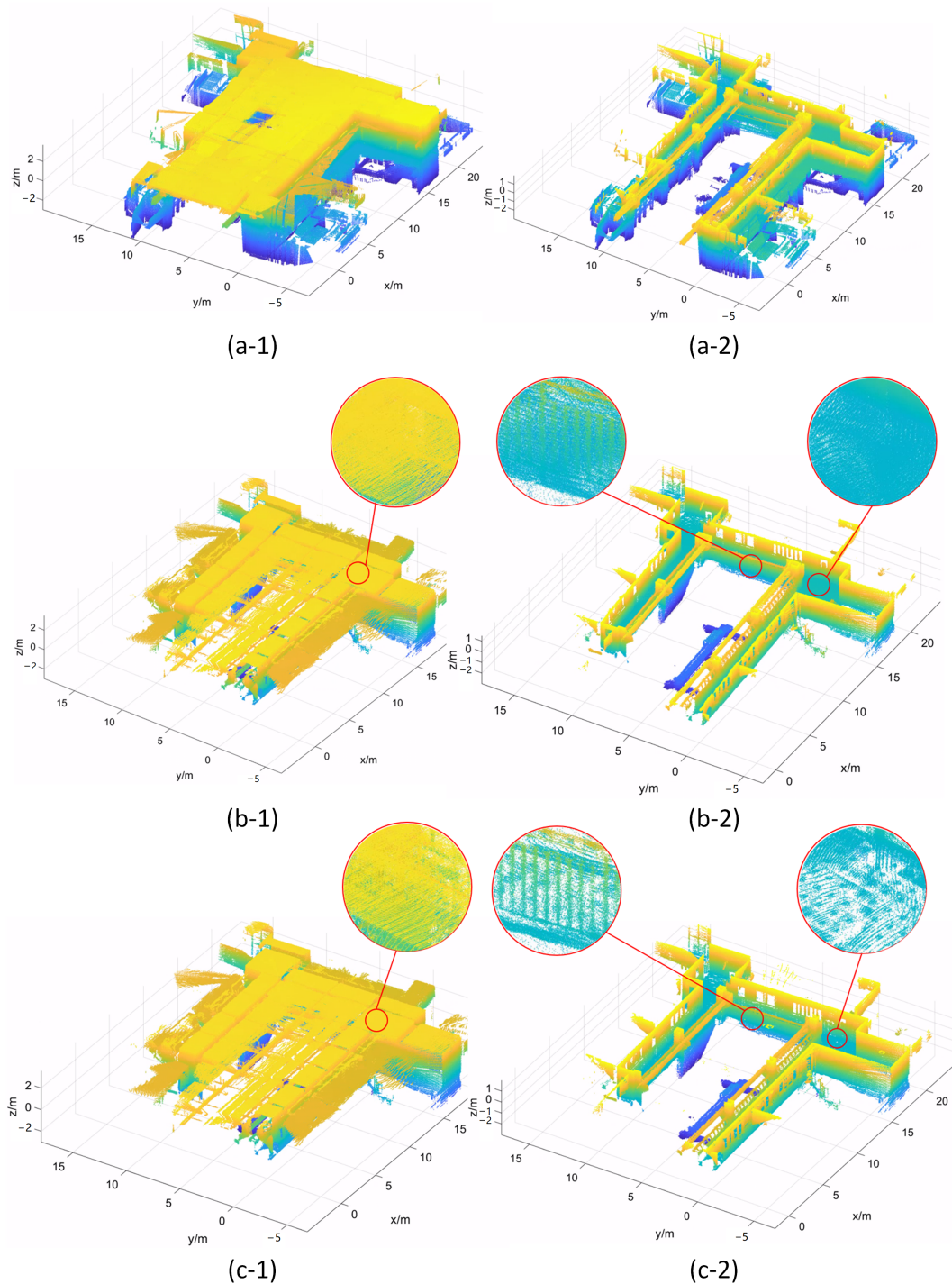


Figure 7. The result of the corridor environment mapping experiment (colored by depth). (a-1,a-2) show the ground truth. (b-1,b-2) show the mapping result without variable resolution strategy. (c-1,c-2) show the mapping result with variable resolution strategy. The full maps are on the left, and the interior details are on the right.

We selected a total of 16 areas (as shown in Figure 8) to calculate the mapping error (as shown in Table 4), among which areas 1–5 are the ceiling, areas 6 and 7 are the floor, areas 8–13 are the railings, and areas 14–16 are the walls. When considering the variable resolution strategy, the mapping errors in these regions are smaller than those without considering the strategy. The accuracy of area 14 is improved most obviously, and the error is reduced by 56.65%.

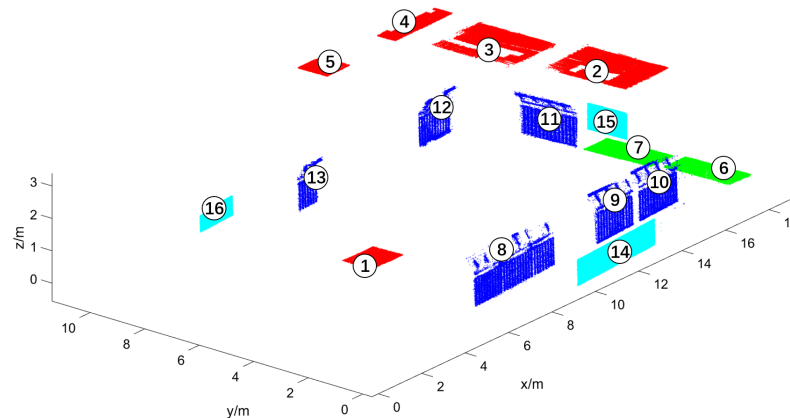


Figure 8. The selected areas of the corridor environment used to calculate the mapping error. Several kinds of regions are selected and colored separately, with each type of region colored in the same color. Areas 1–5 are the ceiling, areas 6 and 7 are the floor, areas 8–13 are the railings, and areas 14–16 are the walls.

Table 4. The mapping error (MAE/m) comparison of the corridor environment.

Area Index	Without Variable Resolution	With Variable Resolution	Area Index	Without Variable Resolution	With Variable Resolution
1	0.0743	0.0694	9	0.0787	0.0627
2	0.0304	0.0284	10	0.0657	0.0435
3	0.0418	0.0394	11	0.0243	0.0217
4	0.0569	0.0503	12	0.0382	0.0283
5	0.0977	0.0973	13	0.0571	0.0560
6	0.0207	0.0188	14	0.0729	0.0316
7	0.0186	0.0171	15	0.0176	0.0167
8	0.0682	0.0573	16	0.0623	0.0444

Smaller errors in bold.

4.2.3. Outdoor Environment

The outdoor environment mapping experiment is carried out in the scene as shown in Figure 9. This scene is a section of the road on the campus. Several cars are parked on the road, many tall trees on the right side, a flower bed on the left, and some short trees on the flower bed.

The environment ground truth is shown in Figure 10(a-1,a-2), the mapping result without considering the variable resolution strategy is shown in Figure 10(b-1,b-2), and the mapping result considering the variable strategy is shown in Figure 10(c-1,c-2). The ground section in Figure 10(a-1,a-2) has several circular vacancies evenly spaced. The reason is that the portable laser scanner cannot scan the area under the device while collecting environmental truth values, and the scanners in adjacent positions are far apart, so they cannot cover each other's scanning vacancies. If adjacent scanings are captured in a small distance, as in the indoor scene, the whole process would be time-consuming and inefficient. Figure 10(c-2) shows the effect of the variable resolution strategy. In the map, the part with small curvature (such as ground) has a small density of point clouds, while the part with large curvature (such as trees and cars) has a large density of point clouds.



Figure 9. The scene of the outdoor environment.

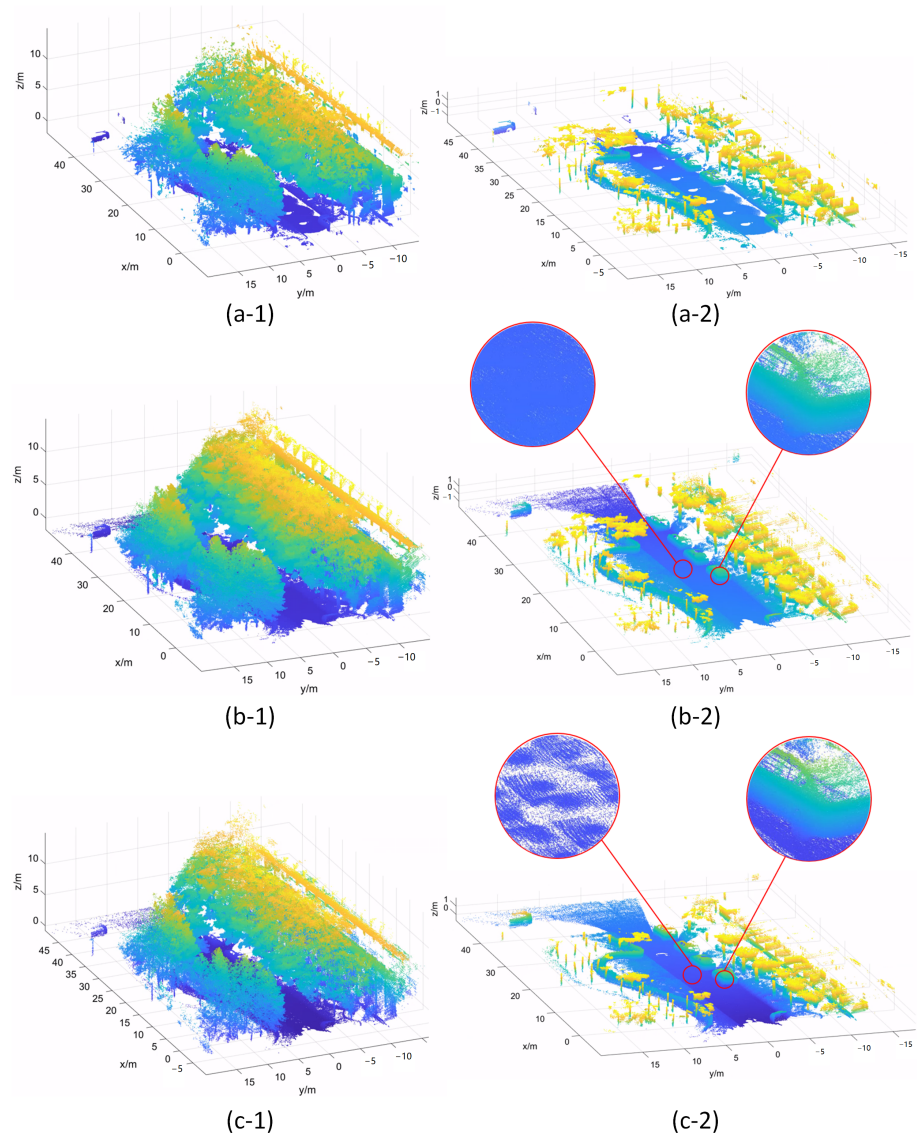


Figure 10. The result of the outdoor environment mapping experiment (colored by depth). (a-1,a-2) show the ground truth. (b-1,b-2) show the mapping result without variable resolution strategy. (c-1,c-2) show the mapping result with variable resolution strategy. The full maps are on the left, and the interior details are on the right.

We selected 30 areas (as shown in Figure 11) to calculate the mapping error (as shown in Table 5), among which areas 1–5 are the ground, areas 6–16 are the flower bed, areas 17–19 are the cars, areas 20–23 are the tall trees, and areas 24–30 are the short trees. The mapping errors of 28 regions with the variable resolution strategy are smaller than those without the strategy. The accuracy of area 4 is improved most obviously, and the error is reduced by 47.50%. In the above three scenes, the most apparent error improvement is in the plane region, that is, the region with small curvature. In the variable resolution map, more abnormal points will be removed from the area with small curvature, so the area with small curvature is more likely to have a significant error improvement.

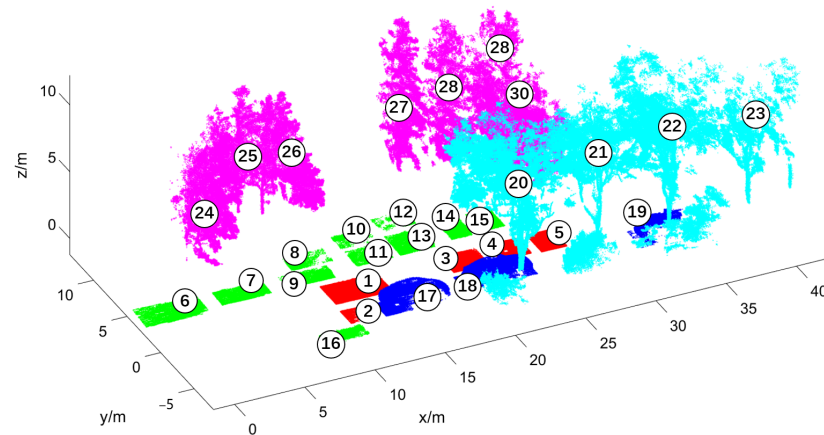


Figure 11. The selected areas of the outdoor environment used to calculate the mapping error. Several kinds of regions are selected and colored separately, with each type of region colored in the same color. Areas 1–5 are the ground, areas 6–16 are the flower bed, areas 17–19 are the cars, areas 20–23 are the tall trees, and areas 24–30 are the short trees.

Table 5. The mapping error (MAE/m) comparison of the outdoor environment.

Area Index	Without Variable Resolution	With Variable Resolution	Area Index	Without Variable Resolution	With Variable Resolution
1	0.0545	0.0417	16	0.0327	0.0327
2	0.0522	0.0416	17	0.0358	0.0356
3	0.0579	0.0517	18	0.0277	0.0287
4	0.0400	0.0210	19	0.0299	0.0290
5	0.0218	0.0194	20	0.0490	0.0485
6	0.0288	0.0254	21	0.0546	0.0541
7	0.0196	0.0188	22	0.0640	0.0630
8	0.0284	0.0261	23	0.0737	0.0735
9	0.0298	0.0269	24	0.0437	0.0423
10	0.0344	0.0340	25	0.0466	0.0453
11	0.0241	0.0234	26	0.0371	0.0359
12	0.0390	0.0382	27	0.0474	0.0456
13	0.0417	0.0374	28	0.0536	0.0516
14	0.0324	0.0294	29	0.0642	0.0621
15	0.0245	0.0218	30	0.0515	0.0499

Smaller errors are in bold.

4.3. Map Size Comparison

In the three experiments in Section 4.2, the comparison of the map size is shown in Table 6. For the room environment, the size of the global map with variable resolution strategy is reduced to 61.02% of that without variable resolution strategy. For the corridor environment, the map size is reduced to 62.67%. For the outdoor environment, the map size is reduced to 79.86%. Because there are relatively few areas with small curvature in the

outdoor environment, the proportion of map size reduction is smaller than in the room and corridor environment.

Table 6. The map size comparison.

Method	Map Size (Point Number)		
	Room	Corridor	Outdoor
without variable resolution	6,173,086	10,355,562	10,986,566
with variable resolution	3,767,088	6,490,328	8,774,072

Smaller numbers are in bold.

4.4. Long-Time Running Experiment

In order to verify the advantages of the variable resolution map update strategy proposed in this paper to build maps in the same space for a long time, but the map does not grow infinitely over time, we conducted repeated mapping for a long time in the experimental scene of a conference room as shown in Figure 12 to cover most details in the scene. This conference room scene has several tables and chairs in the middle of the room and some chairs at the side of the room. The roof of the conference room is a strip multi-storey non-planar structure.



Figure 12. The scene of a conference room.

The environment ground truth is shown in Figure 13(a-1,a-2), the mapping result without considering the variable resolution strategy is shown in Figure 13(b-1,b-2), and the mapping result considering the variable strategy is shown in Figure 13(c-1,c-2). Since there are many flat parts in this environment (for example, the ground and walls), most of the point cloud density is small after the variable resolution strategy is applied, as shown in Figure 13(c-1,c-2). Since the ceiling in this environment is not a planar structure, the curvature is relatively high, and the point cloud density is also rather large when the variable resolution strategy is used, as shown in Figure 13(c-1).

We selected 18 areas (as shown in Figure 14) to calculate the mapping error (as shown in Table 7), among which areas 1–6 are the ceiling, areas 7–9 are the floor, areas 10 and 11 are the walls, areas 12–15 are the chairs, and areas 16–18 are the tables. The mapping errors of 17 regions are smaller when the variable resolution strategy is considered compared with when the strategy is not considered.

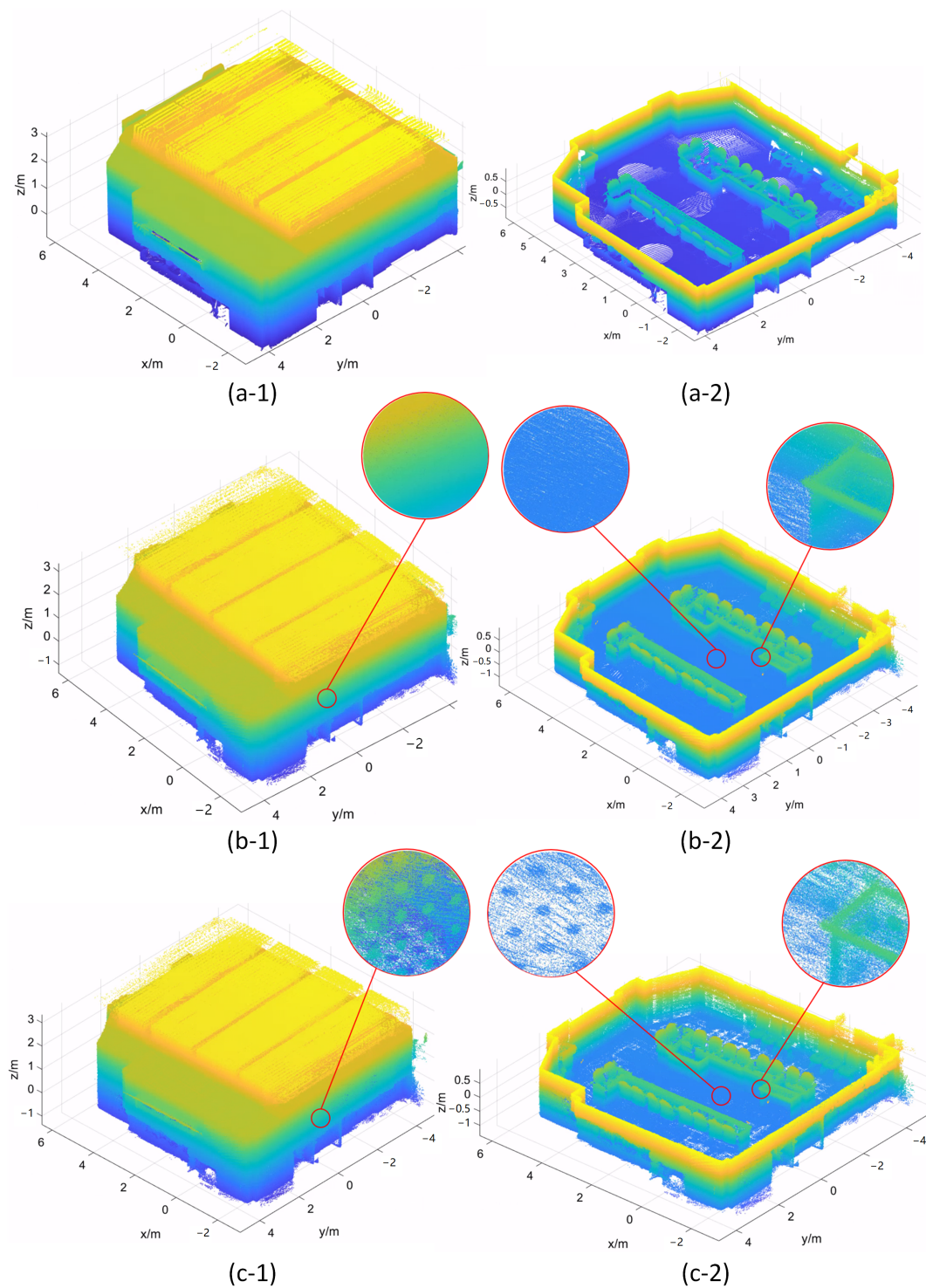


Figure 13. The result of the conference room environment mapping experiment (colored by depth). (a-1,a-2) show the ground truth. (b-1,b-2) show the mapping result without variable resolution strategy. (c-1,c-2) show the mapping result with variable resolution strategy. The full map is on the left and the interior details are on the right.

The size of the maps obtained by the two algorithms is 17,677,016 (without variable resolution strategy) and 6,438,981 (with variable resolution strategy), respectively. The size of the global map with the variable resolution strategy is reduced to 36.43% of that without the variable resolution strategy. The data reduction is expected to be more significant if the experiment time is further extended.

In the mapping process of the two algorithms, the comparison of the size of the global map over time is shown in Figure 15. When we consider using the variable resolution strategy on the map during map construction, the size of the global map will almost stop growing after the scope of the map covers the experimental space rather than increase infinitely with time, as shown in the blue curve in Figure 15. When the variable resolution strategy is not considered for the map construction, the size of the global map will increase infinitely over time, as shown in the red curve in Figure 15. This advantage of the variable resolution strategy proposed in this paper is suitable for long-time mapping because the map size grows with space rather than time.

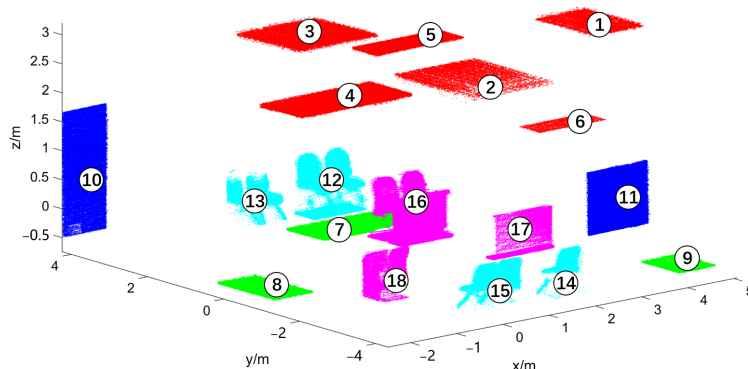


Figure 14. The selected areas of the conference room environment used to calculate the mapping error. Several kinds of regions are selected and colored separately, with each type of region colored in the same color. Areas 1–6 are the ceiling, areas 7–9 are the floor, areas 10 and 11 are the walls, areas 12–15 are the chairs, and areas 16–18 are the tables.

Table 7. The mapping error (MAE/m) comparison of the conference room environment.

Area Index	Without Variable Resolution	With Variable Resolution	Area Index	Without Variable Resolution	With Variable Resolution
1	0.0245	0.0231	10	0.0083	0.0074
2	0.0147	0.0129	11	0.0072	0.0057
3	0.0153	0.0138	12	0.0148	0.0125
4	0.0049	0.0043	13	0.0104	0.0076
5	0.0110	0.0102	14	0.0088	0.0075
6	0.0075	0.0070	15	0.0101	0.0092
7	0.0140	0.0119	16	0.0055	0.0107
8	0.0075	0.0064	17	0.0092	0.0088
9	0.0177	0.0153	18	0.0078	0.0073

Smaller errors are in bold.

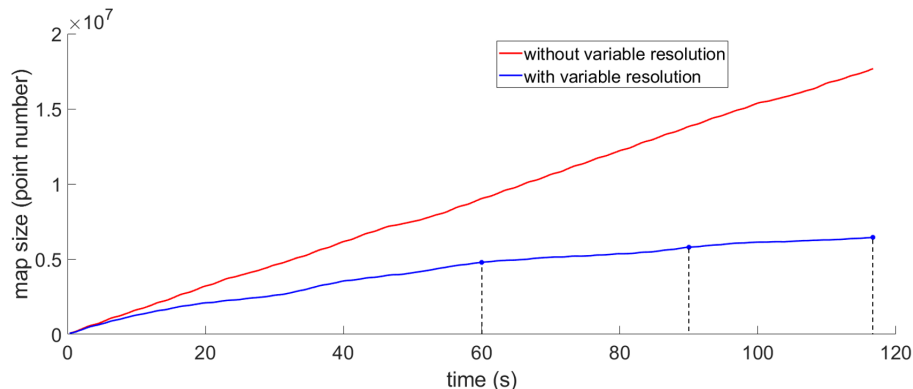


Figure 15. The comparison of map size changes over time under two algorithms.

Figure 16 shows the mapping result of the three time points in the blue line in Figure 15. Figure 16a represents the mapping result at 60 s, at which most of the information in the scene was already constructed. Figure 16b shows the mapping result at 90 s, at which all the scene's information was almost constructed. Figure 16c illustrates the mapping result at the last moment, at which the detailed information in the scene was further improved. After the 90 s, the scope of map construction covered all the space in the scene, so the map size almost stayed the same.

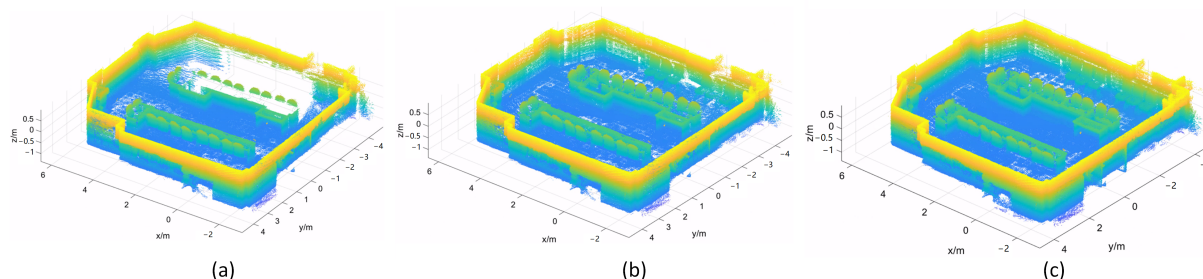


Figure 16. The result of the conference room environment mapping experiment with variable resolution strategy in three time points (colored by depth). (a) shows the result at 60 s. (b) shows the result at 90 s. (c) shows the result at the last moment.

5. Conclusions

In order to solve the problem that the global map grows infinitely with time even in a fixed space in the process of 3D LiDAR mapping, this paper proposes a 3D LiDAR mapping algorithm based on a scan-to-map and NDT registration algorithm, which adopts a variable resolution strategy when updating the global map. The density of the point cloud in the global map adaptively changes with the curvature of different positions. Using this strategy, the size of the map can be reduced with the premise of ensuring the accuracy of the map construction so that the global map changes with space instead of growing infinitely with time.

Experiments in a room environment, corridor environment, outdoor environment, and conference room environment verify the effectiveness of the LiDAR mapping algorithm.

In the future, we may study LiDAR mapping in large-scale dynamic environments.

Author Contributions: Conceptualization, Y.F. and Y.X.; methodology, Y.F. and Y.X.; software, Y.F., Z.G. and J.Z.; validation, Y.F., Z.G. and J.Z.; formal analysis, Y.F., Z.G. and J.Z.; investigation, Y.F. and Z.G.; resources, Y.F. and Z.G.; writing—original draft preparation, Y.F.; writing—review and editing, Y.X.; visualization, Y.F.; supervision, Y.X.; project administration, H.S. and Y.X.; funding acquisition, H.S. and Y.X. 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 62173220 and Natural Science Foundation of Shanghai, grant number 20ZR1419100.

Data Availability Statement: Not applicable.

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

References

- Potena, C.; Khanna, R.; Nieto, J.; Siegwart, R.; Nardi, D.; Pretto, A. AgriColMap: Aerial-ground collaborative 3D mapping for precision farming. *IEEE Robot. Autom. Lett.* **2019**, *4*, 1085–1092. [[CrossRef](#)]
- Huang, J.; Stoter, J.; Peters, R.; Nan, L. City3D: Large-Scale Building Reconstruction from Airborne LiDAR Point Clouds. *Remote Sens.* **2022**, *14*, 2254. [[CrossRef](#)]
- Chatziparaschis, D.; Lagoudakis, M.G.; Partsinevelos, P. Aerial and ground robot collaboration for autonomous mapping in search and rescue missions. *Drones* **2020**, *4*, 79. [[CrossRef](#)]
- Pierzchała, M.; Giguère, P.; Astrup, R. Mapping forests using an unmanned ground vehicle with 3D LiDAR and graph-SLAM. *Comput. Electron. Agric.* **2018**, *145*, 217–225. [[CrossRef](#)]

5. Lombard, C.D.; van Daalen, C.E. Stochastic triangular mesh mapping: A terrain mapping technique for autonomous mobile robots. *Robot. Auton. Syst.* **2020**, *127*, 103449. [[CrossRef](#)]
6. Karachaliou, E.; Georgiou, E.; Psaltis, D.; Stylianidis, E. UAV for mapping historic buildings: From 3D modelling to BIM. *Remote Sens. Spat. Inf. Sci.* **2019**, *42*, 397–402. [[CrossRef](#)]
7. Wang, K.; Ding, W.; Shen, S. Quadtree-accelerated real-time monocular dense mapping. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1–9.
8. Pan, Z.; Hou, J.; Yu, L. Optimization algorithm for high precision RGB-D dense point cloud 3D reconstruction in indoor unbounded extension area. *Meas. Sci. Technol.* **2022**, *33*, 055402. [[CrossRef](#)]
9. Le, T.; Gjevestad, J.G.O.; From, P.J. Online 3D mapping and localization system for agricultural robots. *IFAC-PapersOnLine* **2019**, *52*, 167–172. [[CrossRef](#)]
10. Jiang, Z.; Zhu, J.; Lin, Z.; Li, Z.; Guo, R. 3D mapping of outdoor environments by scan matching and motion averaging. *Neurocomputing* **2020**, *372*, 17–32. [[CrossRef](#)]
11. Kühner, T.; Kümmerle, J. Large-scale volumetric scene reconstruction using lidar. In Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 31 May–31 August 2020; pp. 6261–6267.
12. Wirges, S.; Stiller, C.; Hartenbach, F. Evidential occupancy grid map augmentation using deep learning. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 668–673.
13. Ho, B.J.; Sodhi, P.; Teixeira, P.; Hsiao, M.; Kusnur, T.; Kaess, M. Virtual occupancy grid map for submap-based pose graph SLAM and planning in 3D environments. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 2175–2182.
14. Tao, Q.; Hu, Z.; Zhou, Z.; Xiao, H.; Zhang, J. SeqPolar: Sequence Matching of Polarized LiDAR Map with HMM for Intelligent Vehicle Localization. *IEEE Trans. Veh. Technol.* **2022**, *71*, 7071–7083. [[CrossRef](#)]
15. Chen, P.; Luo, Z.; Shi, W. Hysteretic mapping and corridor semantic modeling using mobile LiDAR systems. *ISPRS J. Photogramm. Remote Sens.* **2022**, *186*, 267–284. [[CrossRef](#)]
16. Koide, K.; Yokozuka, M.; Oishi, S.; Banno, A. Globally consistent 3D LiDAR mapping with GPU-accelerated GICP matching cost factors. *IEEE Robot. Autom. Lett.* **2021**, *6*, 8591–8598. [[CrossRef](#)]
17. Sun, L.; Yan, Z.; Zaganidis, A.; Zhao, C.; Duckett, T. Recurrent-octomap: Learning state-based map refinement for long-term semantic mapping with 3-d-lidar data. *IEEE Robot. Autom. Lett.* **2018**, *3*, 3749–3756. [[CrossRef](#)]
18. Cong, Y.; Chen, C.; Yang, B.; Li, J.; Wu, W.; Li, Y.; Yang, Y. 3D-CSTM: A 3D continuous spatio-temporal mapping method. *ISPRS J. Photogramm. Remote Sens.* **2022**, *186*, 232–245. [[CrossRef](#)]
19. Ibrahim, M.; Akhtar, N.; Jalwana, M.A.; Wise, M.; Mian, A. High Definition LiDAR mapping of Perth CBD. In Proceedings of the 2021 Digital Image Computing: Techniques and Applications (DICTA), Gold Coast, Australia, 29 November–1 December 2021; pp. 1–8.
20. Yuan, C.; Xu, W.; Liu, X.; Hong, X.; Zhang, F. Efficient and probabilistic adaptive voxel mapping for accurate online lidar odometry. *IEEE Robot. Autom. Lett.* **2022**, *7*, 8518–8525. [[CrossRef](#)]
21. Wang, Y.; Funk, N.; Ramezani, M.; Papatheodorou, S.; Popović, M.; Camurri, M.; Fallon, M. Elastic and efficient lidar reconstruction for large-scale exploration tasks. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 5035–5041.
22. Wang, Y.; Ramezani, M.; Mattamala, M.; Fallon, M. Scalable and Elastic LiDAR Reconstruction in Complex Environments Through Spatial Analysis. In Proceedings of the 2021 European Conference on Mobile Robots (ECMR), Bonn, Germany, 31 August–3 September 2021; pp. 1–8.
23. Kim, G.; Kim, A. LT-mapper: A modular framework for lidar-based lifelong mapping. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 23–27 May 2022; pp. 7995–8002.
24. Tu, C.; Takeuchi, E.; Carballo, A.; Takeda, K. Point cloud compression for 3d lidar sensor using recurrent neural network with residual blocks. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 3274–3280.
25. Sun, X.; Ma, H.; Sun, Y.; Liu, M. A novel point cloud compression algorithm based on clustering. *IEEE Robot. Autom. Lett.* **2019**, *4*, 2132–2139. [[CrossRef](#)]
26. Gu, S.; Hou, J.; Zeng, H.; Yuan, H.; Ma, K.K. 3D point cloud attribute compression using geometry-guided sparse representation. *IEEE Trans. Image Process.* **2019**, *29*, 796–808. [[CrossRef](#)]
27. Wang, Q.; Jiang, L.; Sun, X.; Zhao, J.; Deng, Z.; Yang, S. An Efficient LiDAR Point Cloud Map Coding Scheme Based on Segmentation and Frame-Inserting Network. *Sensors* **2022**, *22*, 5108. [[CrossRef](#)]
28. Ryde, J.; Hu, H. 3D mapping with multi-resolution occupied voxel lists. *Auton. Robot.* **2010**, *28*, 169–185. [[CrossRef](#)]
29. Droschel, D.; Stückler, J.; Behnke, S. Local multi-resolution representation for 6D motion estimation and mapping with a continuously rotating 3D laser scanner. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 5221–5226.
30. Sun, Y.; Li, Z.; Yang, Z.; Shao, K.; Chen, W. Motion model-assisted GNSS/MEMS-IMU integrated navigation system for land vehicle. *GPS Solut.* **2022**, *26*, 1–15. [[CrossRef](#)]

31. Meagher, D.J. *Octree Encoding: A New Technique for the Representation, Manipulation and Display of Arbitrary 3-d Objects by Computer*; Electrical and Systems Engineering Department Rensselaer Polytechnic Institute Image Processing Laboratory: New York, NY, USA, 1980.
32. Maurer, W.D.; Lewis, T.G. Hash table methods. *ACM Comput. Surv.* **1975**, *7*, 5–19. [[CrossRef](#)]
33. Hess, W.; Kohler, D.; Rapp, H.; Andor, D. Real-time loop closure in 2D LIDAR SLAM. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1271–1278.
34. Magnusson, M. *The Three-Dimensional Normal-Distributions Transform: An Efficient Representation for Registration, Surface Analysis, and Loop Detection*; Örebro University: Örebro, Sweden, 2009.