https://doi.org/10.20965/jrm.2023.p0470

Paper:

A Map Creation for LiDAR Localization Based on the Design Drawings and Tablet Scan Data

Satoshi Ito, Ryutaro Kaneko, Takumi Saito, and Yuji Nakamura

Gifu University
1-1 Yanagido, Gifu 501-1193, Japan
E-mail: ito.satoshi.s0@f.gifu-u.ac.jp

[Received September 24, 2022; accepted November 22, 2022]

This paper proposes a method for the point cloud data (PCD) map creation for the 3D LiDAR localization. The features of the method include the creation of a PCD map from a drawing of the buildings and partial scan of the not-existing object of the map by the tablet computer with the LiDAR. In the former, a map creation procedure, including the up- and down-sampling, as well as the processing, with voxel grid filter is established. In the latter, automatic position correction of the tablet scan data is introduced when they are placed to the current PCD map. Experiments are conducted to determine the size of the voxel grid filter and prove the effect of the tablet scan data in enhancing the matching level and the localization accuracy. Finally, the experiment with an autonomous mobile robot demonstrates that a map created using the proposed method is sufficient for autonomous driving without losing the localization.

Keywords: self-localization, point cloud data, map creation, design drawings, tablet computer

1. Introduction

Cost reduction in the manufacture of low-price products based on inexpensive labor is a promising solution to enhancing the international competitiveness in the international markets. The production technologies will contribute to it. Shorting the production time increases the amount of the production per hour. Increasing the quality of the production indirectly reduces the production cost with less defective pieces. Therefore, the well-designed automation technologies will largely improve the productivity by replacing the human workings with the machine operations. Machines can endure the long-term labors with suppressing the employment costs. Among the automation technologies, this paper mainly treats that of the transportation, especially in the factories of job-shop type, adopted, for example, in the airplane industry.

Industrial factories require the transportation of parts in its assembling process. In the factories with the conveyor system, the machines for producing, processing, or assembling are installed round the conveyors. These machines are fixed in the factories; thus, the factory layout does not change frequently. In such a situation, autonomous guided vehicles (AGV) that use a "guide," such as the rails, magnetic devices, or simply painted lines on the floor, are available and widely adopted to convey the parts. The AGVs use these guides to detect their current location in the factory and travel along the guide to their destination.

However, some factories, such as airplane industry, have different situations. At a so-called job-shop type factory, the products, such as planes, are not placed on the moving conveyor; their positions are fixed, but the assembling parts and the power tools move. As a result, the factory layout changes frequently during the assembly process. The variable factory layout is poorly compatible with the AGV: the products, parts, power tools, or even workers often change their position; thus, the path of the AGV might be modified depending on their position. The variable factory layout makes it difficult to equip the fixed guide for AGV.

In addition to such a guide, factories do not want to newly set up some other sensors, such cameras or beacons, because of the cost, maintenance, and installation, which could disrupt the factory's operation. These constraints limit the usage of sensors to those built inside the autonomous mobile robot (AMR). As a result, here, we introduce the 3D LiDAR sensing technology, which is utilized in autonomous driving cars. This sensing technology allows cars to run on the road of our social living space without the use of rails, as trains do.

The 3D LiDAR localization requires a map of the precise moving space, which is usually given as the point cloud data (PCD) map. With the map, the AMR can estimate their current position in this space. The methods that can create the map with localization, such as simultaneous localization and mapping (SLAM), have already been proposed [1–3]. Indeed, SLAM technology is certainly effective for exploring completely new environments. However, the moving space for the AMR is the factory, which is a restricted and partially known place. Under these circumstances, the map must be prepared in advance to reduce the computational cost for creating it during automatic transportation. From now on, this paper will focus on the map creation for the 3D LiDAR localization.

The 3D LiDAR on the AMR is sometimes utilized to scan the environment and mapping [4]. The utilization of the 3D LiDAR on the AMR has some merits: we do not have to purchase new equipment; we can use the AMR we already have; the same 3D LiDAR is used for localization; thus, the data is free from the individual property differences in the sensors. In the mapping and localization process, the condition on the data acquisition, particularly in the height of the LiDAR position, is the same. However, when we introduce this method to the actual factory, some factory staffs have to learn both how to operate the AMR to scan the factory environment and how to create maps from the scanned data. This kind of the education has become a major issue in the actual factory. Namely, a map creation without the scan with AMR is preferable.

Many studies on map creation, such as point cloud mapping [5], have been published. One of the powerful methods this paper utilizes is the normal distribution transformation (NDT), which divides the mapping space into several pixel/voxel grids of the same size and approximates the distribution of point clouds using the continuous Gaussian function in each grid [6, 7]. This approximation with the continuous Gaussian function enables us to compute matching computation using derivatives, though the discretized point cloud data are originally non-differentiable. The NDT approach is improved in some ways, such as adaptive resolution [8] and combining the occupancy maps [9]. However, this paper focuses on the data before it is transformed: how to create the feature map from point clouds for the localization without using a LiDAR scan.

To the field of the factory automation, the localization technology is recently being introduced. Yilmaz et al. [10] utilized affine interactive closet point method for the localization in the manufacturing systems with introducing the correlation entropy criterion. Zhou et al. [11] combined the point cloud data with the 2D camera information to detect the localization of the many tools' part to handle with the mobile manipulator based on the deep learning. Our aim is also the factory automation, but the approach is different in that our focus is placed to map creation for the LiDAR localization. There, we discovered that the place the AMR travels is almost indoor. If so, a drawing of the interior of the building should be available for indoor navigation [12, 13]. Although the scale might be different from the actual one, the relative size or positional relation between rooms in the drawings will be precise. One idea in this paper is to utilize the figure information on the drawings to create the map for the 3D LiDAR localization. This will allow us not to visit the factory for the map acquisition, as Takahashi et al. [14] focused on its merit. Hoshino and Yagi [15] utilized the cadastral map for the outdoor navigation of AMR. Though our idea might be similar to theirs, the target and thus the approach becomes different in whether the PCD map or the occupancy grid map. However, there would be some objects not being drawn in the factories, such as the power tools and containers. Another idea in this paper is to scan non-existing objects in the drawings using a portable LiDAR, i.e., the tablet terminal with the LiDAR, never the LiDAR on the AMR. This idea can cope with the case where the factory layout is partially changed, i.e., variable factory layout. Many objects are modeled as the PCD [16]. The major features of this paper are that design drawings and a widely used tablet computer with LiDAR are used to create the PCD map, and this map is applied for the autonomous driving of an AMR. Thus, the aim of this paper is different from the study of indoor point clouds for reconstructing a parametric 3D building models [17], the classification [18], or the semantic segmentation [19–21]. The related works are summarized on **Table 1**.

The remainder of this paper is organized as follows. Section 2 establishes a process to create the map for the 3D LiDAR localization from the drawings and to add the tablet scan data on the map. In this map creation process, some parameters have to be chosen. Sections 3, 4, and 5 are devoted to the experimental studies: Section 3 demonstrates not only a parameter selection relating the down-samplings, but also the localization improvements by tablet scan data. Section 4 introduces the automatic positional correction of the tablet scan data when it is placed to the map created from the drawings, and the allowable deviation is experimentally investigated. And, Section 5 aims to achieve the autonomous driving using the map created by our method. Finally Section 6 concludes this paper.

2. Map Creation for 3D LiDAR Localization

2.1. Purpose

The main purpose of this paper is to create a map for 3D LiDAR localization. This paper adopts the algorithm "NDT matching" for the localization [22]. The NDT matching treats the data from both the LiDAR scan and the map as the PCD. Therefore, a 3D map has to be created in the PCD format, in which the 3D space where the walls or objects exist is filled up with the point data.

2.2. PCD Map Creation from Drawings

Here, we propose a method to create the 3D map in the PCD format (hereafter, "PCD map") from the 2D drawings. In this method, we will utilize the geographical information, such as the relative position or size of the rooms, corridors, walls, or doors in the drawings.

We established a method to create the PCD map [23] considering the following factors.

a. Discretization: The transforming of drawings to the PCD map is a kind of spatial discretization: the continuous lines representing some walls or partitions within the buildings have to be expressed by a huge amount of discrete point data, i.e., point clouds. At the current estimation position, too few points cannot adequately describe the environments, while too many points consume the computational cost at the current position estimation. Thus, proper down-sampling is required. For this down-sampling, we apply the voxel grid filter [a] to distribute

Table 1. Related works.

Method	Characteristics	Our method
SLAM (e.g., [1–3])	Map creation with localization. Large computational cost.	can create map from design drawing before bringing AMR to the cite.
NDT (e.g., [6–9])	Utilization of continuous Gaussian function to express distribution of discrete PCD. PCD is usually obtained from LiDAR on AMR.	also utilizes NDT, but PCD data is obtained from LiDAR on NOT AMR BUT tablet computer.
Yilmaz et al. [10]	Affine interactive closet point method introducing correlation entropy criterion. Target is manufacturing system.	adopts NDT and target is factory mapping.
Zhou et al. [11]	PCD plus 2D camera information is utilized for mobile manipulator. Target is tools' part localization.	aims at map creation for self-localization only from PCD.
Pinto et al. [12]	Localization algorithm for indoor navigation using camera image and tilting laser range finder.	treats same indoor navigation, but target is PCD map creation.
Zhang et al. [13]	Indoor localization from wheel odometry, inertia measurement unit and ultra wideband by without LiDAR and GPS.	treats same indoor navigation, but target is PCD map creation.
Takahashi et al. [14]	Outdoor localization based on edge-node map using inner sensor.	treats LiDAR indoor localization based on PCD map.
Hoshino and Yagi [15]	AMR outdoor navigation based on occupancy map and cadastral data.	is for indoor navigation based on PCD map from design drawings.
Studies on PCD [16–21]	PCD are utilized to 3D modeling, model reconstruction, classification, semantic segmentation and so on.	applies PCD to localization.

the points uniformly with almost the same intervals.

b. Height information: 3D PCD map should contain the height information, but the 2D drawings do not. This solution to this problem is to construct adequate high walls in the PCD map. Most of the radar from the LiDAR reflects to the wall and does not reach the ceiling because the AMR moves indoor, and thus, the LiDAR scan distance is short.

c. Scaling: The scale varies depending on the drawing. To scale the map to the actual distance, we execute the up- or down-sampling of the PCD map according to the scale of the drawings at the final process.

The procedure of the map creation from the drawings that we propose is summarized as follows:

1. *Input the drawing to the computer.* Suppose that the drawings are given in the paper. The paper data are first scanned and then inputted to the computer. The binary image data is then obtained.
2. *Preprocessing the image.* The extra information other than the walls, partitions, such as the dimension lines or the numbers, are eliminated. Then, the voxel grid filter is applied to down-sample the image. At this stage, the 2D discretized map is obtained.
3. *Generating the 3D map.* Following the PCD format, the 3D PCD map is generated based on the map information. It is a process to accumulate the point cloud data to the height direction at the position of the walls or partitions.
4. *Rescale.* At the last stage, the scale of the PCD map is adjusted to match the real space.

Fig. 1 shows a CPD map created from the drawings. The PCD were distributed uniformly along the wall surfaces, with the constant gap size, as we expected thanks to the voxel grid filter. However, the rate of the down-sampling in relation to the point cloud density is an essential problem. We will examine it experimentally in Section 3.

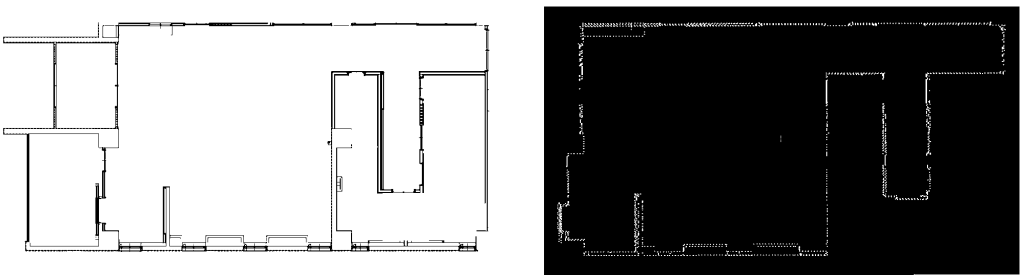
2.3. Non-Existing Objects on the Drawings

Actual factories have many objects not included in the drawings, such as power tools, containers, assembly parts, and products. Furthermore, these objects tend to change their position. Under these conditions, the map have to be updated up to the current situation, before operating the AMR.

Here, we propose a method for updating maps using portable tablet computers with a LiDAR. Indeed, it is a demerit that the tablet computer has to be newly purchased: if the LiDAR equipped on the AMR is utilized, the cost for new sensors is unnecessary. In contrast to the equipped LiDAR, however, the tablet computer has some usages, such as social media, games, or music player, other than the sensor, and its human-interface is excellent. Thus, we concluded that some of the users might have already possessed the tablet computer or find it worthwhile to get one. Throughout this paper, "tablet scan" means an action to obtain the PCD data of the real world using the tablet computer with the LiDAR, and the obtained PCD data is called "tablet scan data."

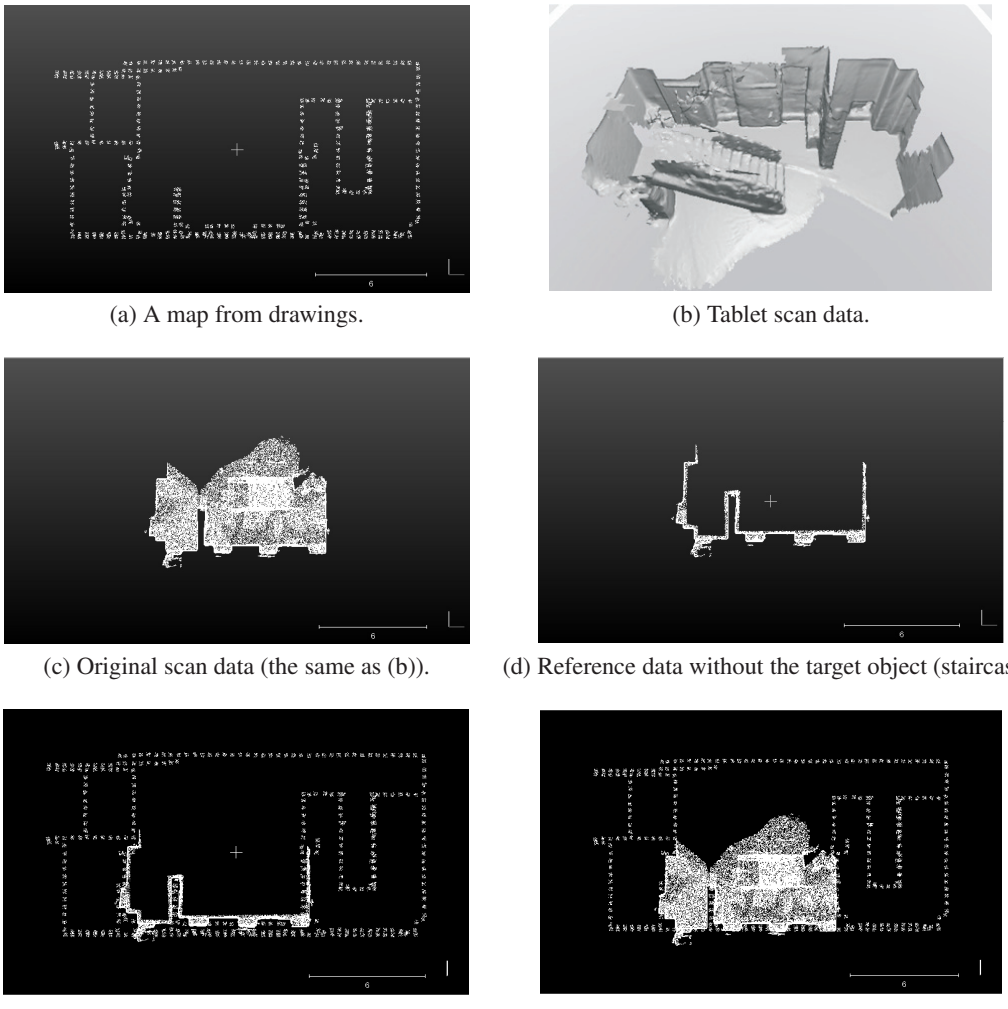
A crucial problem in this idea is how to place the scan data at the appropriate position on the PCD map. A free application software, "Cloud Compare," allows us to graphically display a series of PCDs in a computer window and alter its position with the mouse device.

In fact, we can place the scan data intuitively to the appropriate position at the PCD map. However, the manual placement with the mouse device seems inexact. As



(a) Drawing. (b) PCD map.

Fig. 1. Example of PCD map created from a drawing.



(a) A map from drawings. (b) Tablet scan data.

(c) Original scan data (the same as (b)). (d) Reference data without the target object (staircase).

(e) Rough placement. (f) Target object at the correct position.

Fig. 2. Automatic placement of tablet scan data.

a result, we believe a function is required such that, when the scan data is placed near the true position on the GUI software, it is automatically placed to its correct position.

To achieve the above idea, we found that the tablet scan data should include the PCD data for not only "the target object," i.e., the missing one that should be newly added to the current PCD map, but also already-existing "the reference objects" working as a key to place the target object in the correct position on the map. Thus we decided to require the users to perform the following operations for the tablet scan:

1. *Tablet scan.* This scan should include the reference objects, such as walls or some other landmarks in the factory that are already on the current PCD map (**Fig. 2(a)**).

2. *Preprocessing.* The target object has no data to match on the current PCD map (**Fig. 2(b)**) since it does not exist there. In other words, the target object data inversely impedes the localization. The match-

ing for the localization will be computed effectively only for the reference objects that are already on the PCD map. Thus, only the reference objects should remain, as seen in **Figs. 2(c)** and **(d)**, by eliminating the target objects. (So far, this eliminating process is executed manually by "Cloud Compare" in which we can select the object to eliminate easily with the mouse operation.)

3. *Rough placing.* To facilitate the convergence to the correct position, the user teaches a rough position of the object. In **Fig. 2(e)**, the reference LCD is placed at a position close to the correct one.

4. *Matching.* Run the matching algorithm between the reference object data and PCD map by setting the position given in the previous step as the initial value. As a result, the coordinate transformation to the estimated correct position is obtained.

5. *Compound.* Translate the position of the target object PCD (which can include the reference data) according to the result of the previous step, and compound it to the current PCD map (**Fig. 2(f)**).

The third step determines the extent the user can deviate from the true position of the target object. This will be examined by the simulation case study in Section 4.

3. Experiment 1: Localization on the Map from Drawings

3.1. Purpose

Although we showed the effectiveness of the voxel grid filter in the map creation in **Fig. 1(b)**, the appropriate size of the filter in the actual application is unknown. Additionally, the effect of the tablet scan data is not confirmed. Thus, the purpose of this section is to empirically examine the following:

- the appropriate size of the voxel grid filter, and
- the effect of the tablet scan data,

while evaluating the localization results for the various maps created in the different conditions.

3.2. Experimental Setup

This paper focuses on the map creation method. Thus, what we should evaluate is the map; the accuracies of the localization were evaluated for the maps we created with the different conditions in the first two experiments. The experimental setup was constructed so that all other conditions, such as the localization algorithm or input data, became the same across all maps.

The experiments were conducted at a building in Gifu University. Accordingly, all of the maps were created from the drawings of this building, as illustrated in **Fig. 1(a)**.

The data for the 3D localization were obtained using the cart shown in **Fig. 3(a)**, which featured a LiDAR (Velodyne LiDAR VLP-16) at the top shelf at a height of 0.8 m from the floor, a personal computer (PC) at the middle shelf, and a direct current (DC) battery at the bottom shelf. The PC was operated by the robot operating system (ROS) [b], on which the free auto-driving control software, "Autoware," [c] was running. In the experimental space, six points are selected for the localization, as shown in **Fig. 3(b)**. These points were marked on the floor with yellow circular seals of 1 cm radius, and the exact position was manually measured using a measuring tape. The cart was placed on the yellow seals so that the LiDAR position corresponded to exact upper position of the seals, and the LiDAR data was collected for 5 s using the ROS command, "rosbag." This operation was repeated at each six position in a predetermined scan order. A series of the six scans comprise one trial, and the six trials were conducted in the different scan order starting at the marker 3. These rosbag data enable us to evaluate the localization results for each map in the same way; the rosbag can recreate the same situation as if exactly the same data were obtained from the LiDAR at the same timing.

3.3. Analysis

An algorithm, NDT matching, was applied to the LiDAR scan data against the map created under the various conditions. It not only calculates the current position estimate in the map, but also an index value. Transformation probability (TP) [24] indicates the matching level between the scan data and the referred map. We evaluated the appropriateness of the created map with the values: TP and the positional error of the estimate from the actual value, which were obtained with a measuring tape.

To compare the matching level of the map created from the drawings, we also examine TP to the map created from the LiDAR scan data, which is often utilized for the map creation. These scan data were acquired from the same LiDAR that was used in the experiments and was installed at the same height. Accordingly, a high TP evaluation was expected to be obtained from this map. In other words, we could improve our map creation method at most to this TP value.

The TP does not indicate the accuracy of the positional estimate. For the six trials, the average of the estimates during the period the cart stopped at each target position was calculated. The average of the six trials, as well as the error ellipse from these six values, were also calculated.

3.4. Effect of the Size of the Voxel Grid Filter

3.4.1. Methods

In pilot experiments, we confirmed that the map created from the drawing can accurately estimate the current position without computation divergence. Here we aimed at clarifying which size of the voxel grid filter provides the best result; therefore, we tested the seven filter sizes: 0.025, 0.05, 0.1, 0.2, 0.3, 0.4 and 0.5 m. **Fig. 1(b)** shows one of the map we created.

(a) Cart with LiDAR. (b) Experimental space.

Fig. 3. Environment and setup for experiment.

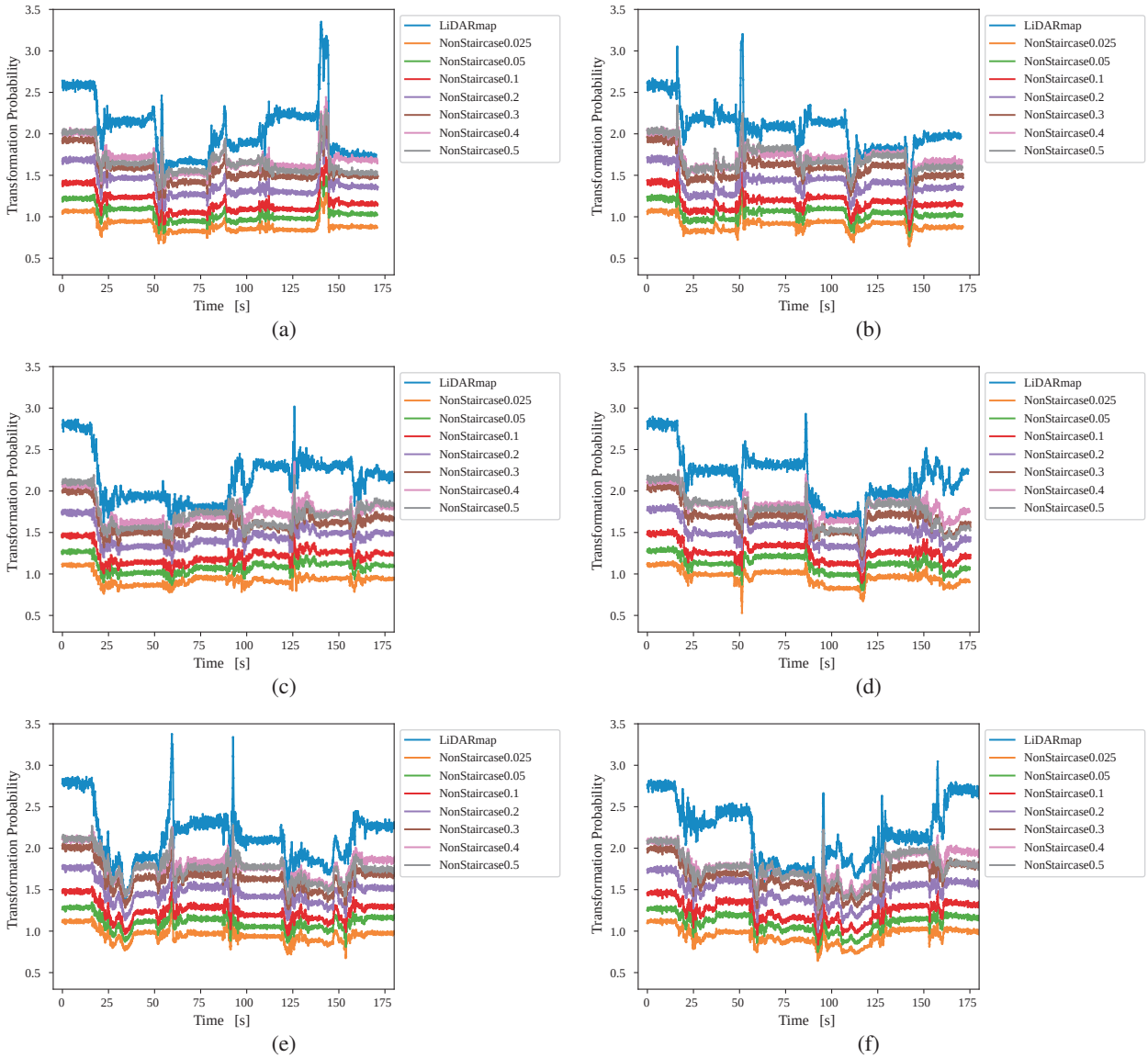


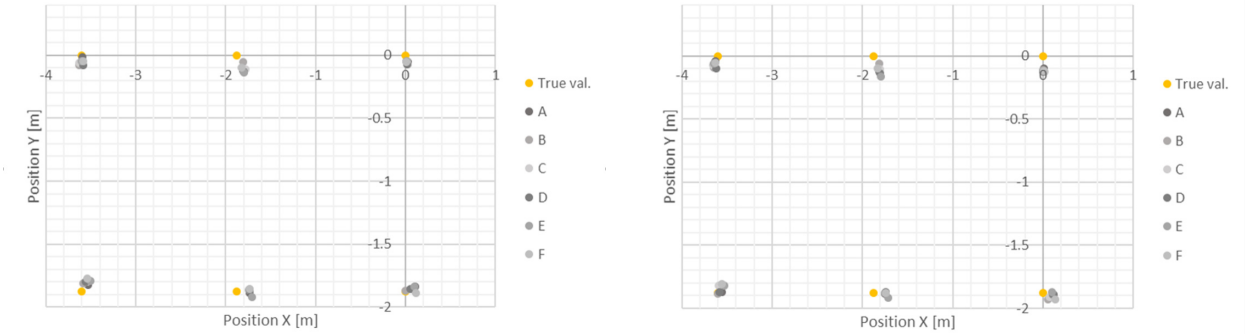
Fig. 4. Comparison of TP values for the different gap size.

3.4.2. Results and Remarks

In **Fig. 4**, the changes of TP are illustrated for each trial. **Fig. 5(a)** shows the average of each target point for each filter size.

From **Fig. 4**, the 0.40-m filter size got the best TP, except for the map from the LiDAR scan. This implies that too large or too short filter size cannot provide the best matching, and that the optimal size exists.

Figure 5(a) shows the positional error for the 0.40-m filter size of the best TP. If the filter size was selected correctly, the error of the estimates was almost within 0.1 m.



(a) Map from drawing only.

(b) Map with tablet scan data.

Fig. 5. Estimated positions.

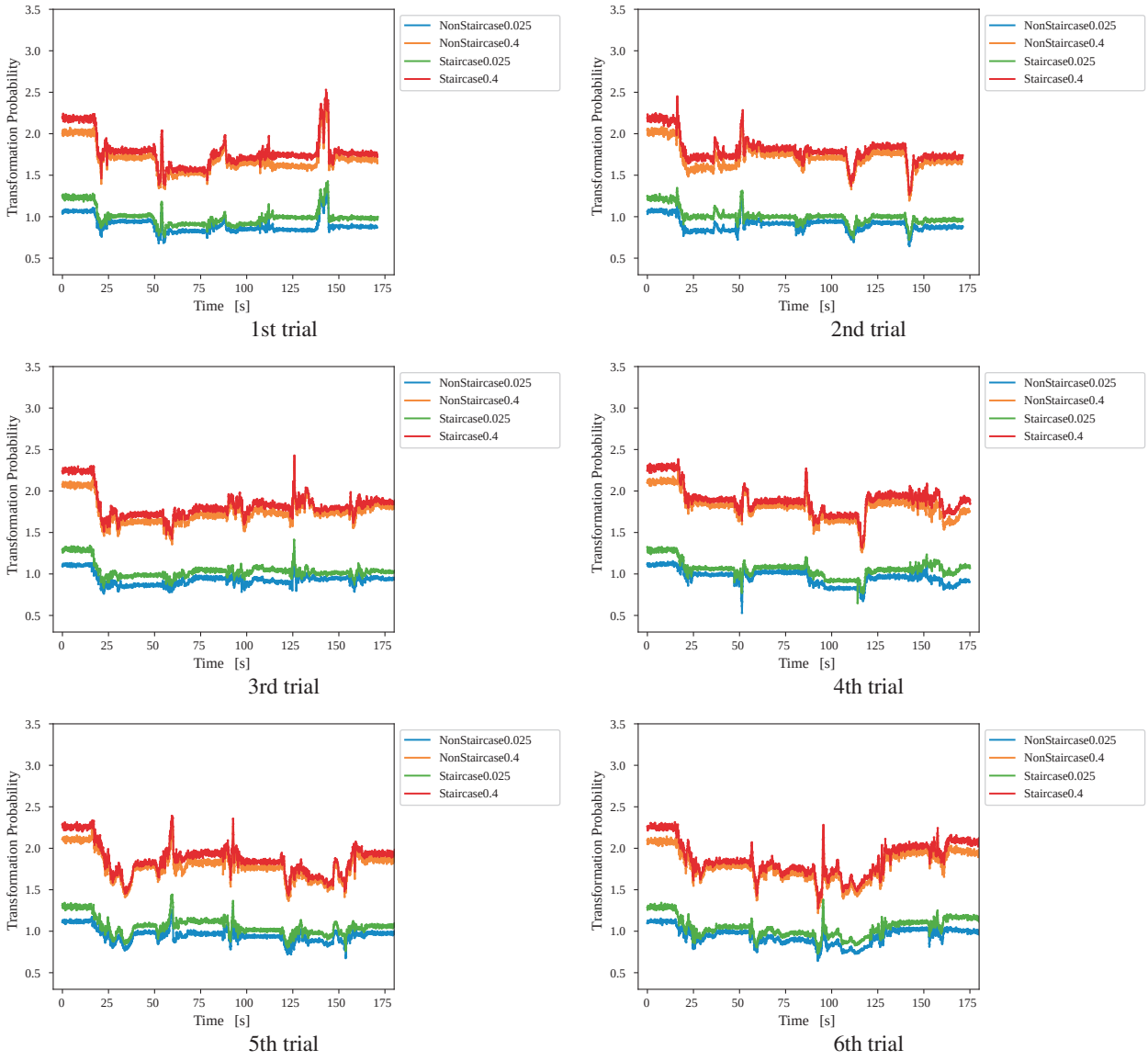


Fig. 6. Comparison of TP with and without the table scan data.

3.5. Improvement of Tablet Scan Data

3.5.1. Methods

The tablet scan data is compounded to some PCD maps from the drawings, following the method in Section 3.4. Exactly the same LiDAR data as Section 3.4 are inputted for the 3D localization using the rosbag command, and the TP is compared with **Fig. 6** at the best filter size of 0.4 m and the worst filter size of 0.025 m.

3.5.2. Results and Remarks

Figure 6 illustrates the comparison of TP for each trial. **Fig. 5(b)** shows the average of each target point for each filter size.

During most of the period, TP is higher in the map with the tablet scan data than in those without it, as shown in **Fig. 6**. Namely, we can conclude that the tablet scan data certainly improves the matching accuracy.

In terms of localization accuracy, the position is estimated to be within 0.13 m error, as shown in **Fig. 5(b)**. In most trials, the error is within 0.1 m on average and is smaller than the map without the tablet scan data. Thus, the tablet scan data should be compounded in terms of accuracy.

4. Experiment 2: Allowable Placement Deviation of Tablet Scan Data

4.1. Purpose

Section 3.5 demonstrated that the tablet scan data improves the localization performance as expected. However, it requires the scan data should be placed at the correct position. The automated adjustment mechanism in Section 2.3 is a convenient way to achieve this easily. However, the algorithm to find the correct position of the tablet scan data, i.e., the matching algorithm, does not always work appropriately due to its local minima in the computation process; to avoid this, the initial position should be placed as close to the true position as possible. Here, we experimentally examine what extent of the deviation is allowable for the manual placement of the tablet scan data.

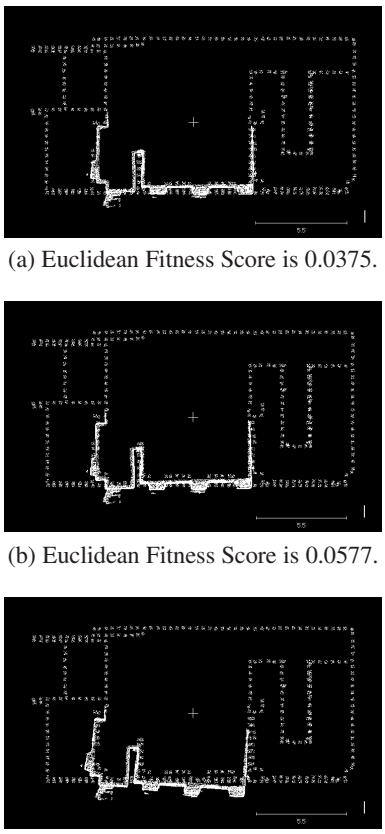
4.2. Methods

The PCD map and tablet scan data from Section 3 are used: the PCD map was created from the drawings of a building at Gifu University containing no staircase. The size of the applied voxel grid filter was 0.4 m. The staircase was scanned using the application software CANVAS (Occipital, Inc.) on iPad pro 2nd generation (Apple Inc.) tablet computers with LiDAR.

The positional and rotational deviation from the correct placement was independently given as the initial position of the tablet scan data for the repetitive computation of the matching algorithm. The positional deviation was measured in the range from -0.9 to $+0.9$ m in all directions with 0.1 m interval, while the rotational deviation was measured in the range from $-15°$ to $+15°$ with $1°$ interval. The center of the rotational deviation was set to the origin of the object PCD, which was the starting point of the tablet scan in the real space.

For the matching algorithm, NDT matching was adopted by calling its function from the C++ library "Point Cloud Library" [d]. The grid size of NDT matching algorithm is selected as 1.0 m.

Euclidean Fitness Score, which is obtained from a function in the Point Cloud Library, is used to evaluate the matching. The Euclidean Fitness Score is a positive number that represents the averaged squared distance between two point clouds, and thus represents the matching level of the two point clouds match. In the pilot test, this score ranges from 0.036 to 0.038 if we can visually ascertain that the object PCD was placed at the appropriate position as shown in **Fig. 7(a)**. However, if the value is over 0.038, we can find an inappropriate matching, such as the placement far from the correct position (**Fig. 7(b)**: estimated lower), or a wrong orientation (**Fig. 7(c)**: estimated clockwise). In the experiment, the Euclidean Fitness Score was over 0.038, implying that the matching was not achieved.



(a) Euclidean Fitness Score is 0.0375.

(b) Euclidean Fitness Score is 0.0577.

(c) Euclidean Fitness Score is 0.1736.

Fig. 7. Result of the pilot test for automated adjustment of the tablet scan data.

4.3. Results

The influence of translated deviation was depicted in **Fig. 8** for the vertical deviation $\Delta z = 0, \pm 0.1, \pm 0.3, \pm 0.5$, and ± 0.9 m. Each figure contains a 19×19 lattice, whose rows correspond to the forward-backward deviation Δy and columns correspond to the lateral deviation Δx in the horizontal direction. Symbols represents the largeness of the final Euclidean Fitness Score: it is smaller than 0.038 at $\bullet$ indicating that the correction was succeeded, between 0.038 and 0.05 at $\circ$, and larger than 0.05 at $\times$ where the result was far from the correct position.

Figure 9 shows the final Euclidean Fitness Scores depending on the initial rotational deviations.

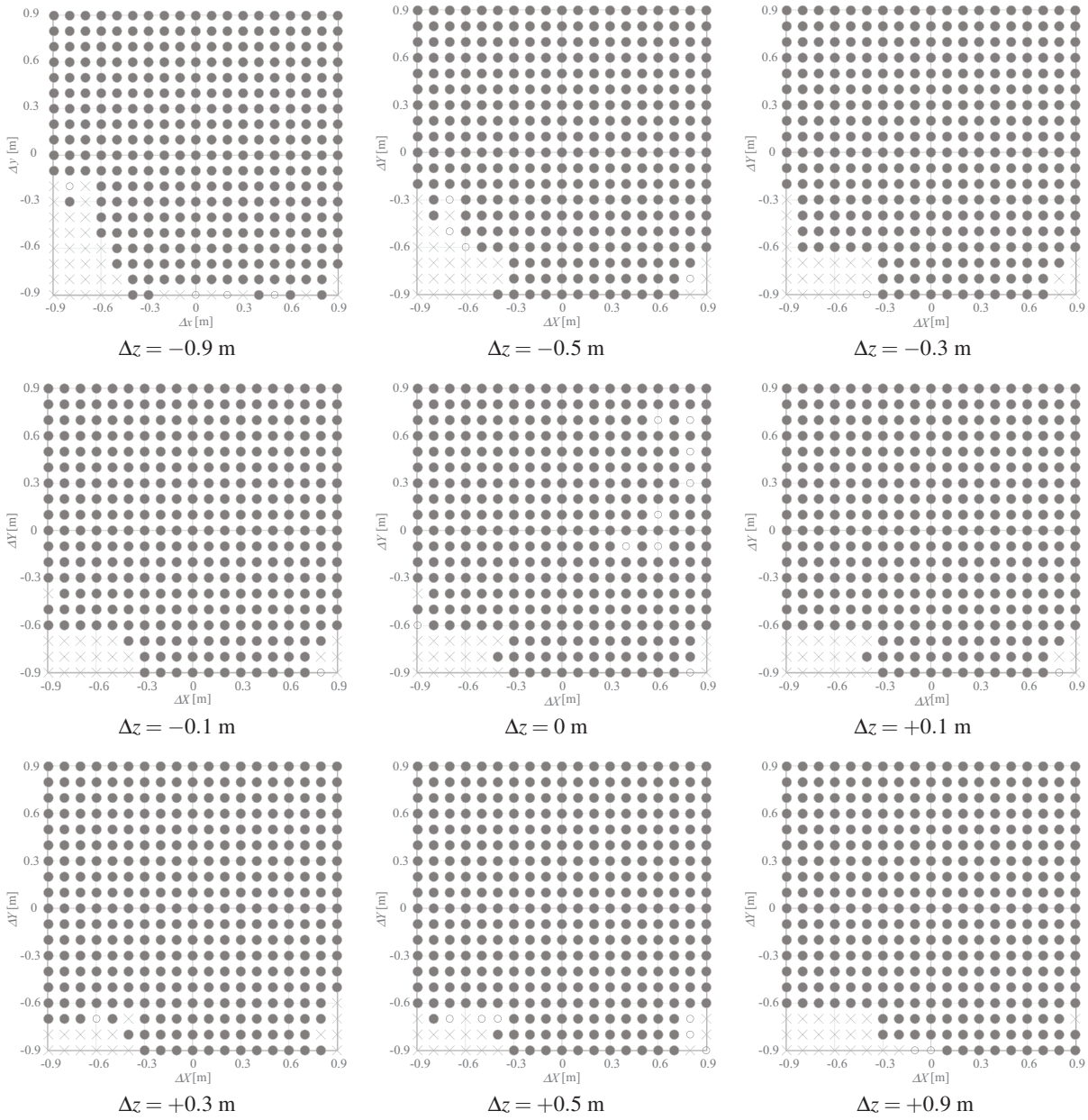


Fig. 8. Final Euclidean Fitness Score depending on the initial translational deviations.

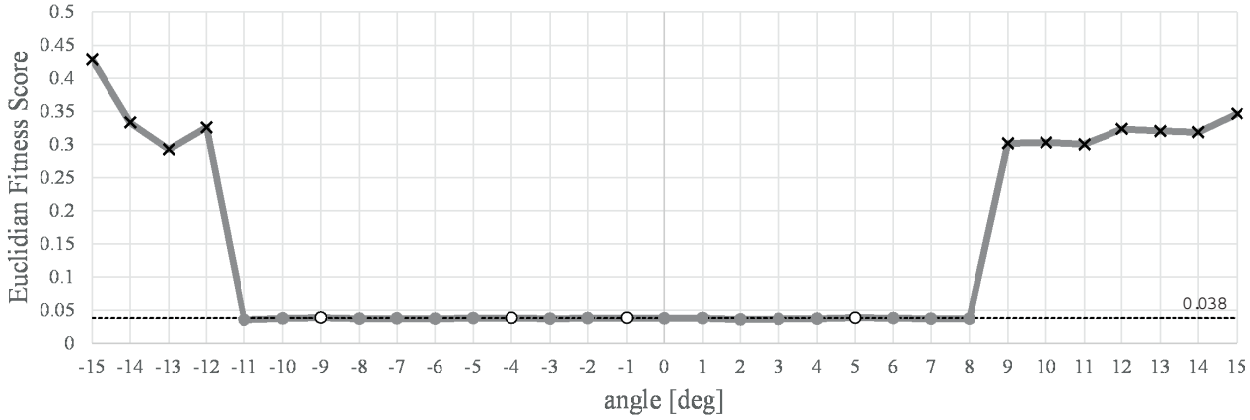


Fig. 9. Final Euclidean Fitness Score on the initial rotational deviations. The meaning of the marks, •, ○ and ×, are the same as **Fig. 8**.

4.4. Remarks

The convergence to the correct position becomes difficult when the initial placement deviates to the negative Z direction. The PCD map created from the drawings did not include neither the ceiling nor the floor, while the staircase PCD included the ceiling but not the floor. Thus, the matching algorithms try to match the ceiling data of the staircase PCD placed lower than actual due to the initial deviation to the negative Z direction, even to the wall in the PCD map. To avoid it, some improvements on the map creation will be required, such as adding the ceiling.

In this experiments, a deviation of more than 0.9 m is allowable in the positive Y direction, even if the deviation in the X direction is equal to 0.9 m. However, if the amount of the deviation is within 0.7 m to the negative Y direction, convergence to the correct position is ensured. The difference in the Y direction will be caused by the width of the wall in the PCD map. The staircase PCD contains only the inside of the wall. Thus, the initial placement to the outside will ensure that the wall in the staircase PCD matches to the outside of wall in the PCD map. This may be avoidable by instructing the users when we designate the initial placement of the object PCD. Consequently, if the object PCD is set to the inner part of the wall, a 1-m deviation will be allowable in its initial placement.

However, the rotational deviation was allowed in the range from $-11°$ to $8°$, where the Euclidean Fitness Score was under 0.05. Outside that range, the Euclidean Fitness Score drastically jumped to over 0.3. Asymmetry in Euclidean Fitness Score to the positive-negative rotation will come from the origin of the rotation and the room structure, such as the wall position. In this case, there are more point clouds in the lower left area in the map. Thus, the negative rightward rotation did not result in a greater reduction in matching points than the leftward rotation.

Comparing the translational and rotational deviation as an initial position of the matching algorithm, the former tends to be more robust. In some other rotational experiments in the different origin, only a one-degree rotation largely changed the matching among the points clouds. Regardless, we should place the initial object PCD allowing few rotational deviation, which can be achieved easily by starting the tablet scan with the normal direction to the reference wall in the real space.

5. Experiment 3: Autonomous Driving

5.1. Purpose

The purpose of this experiment is to confirm that the map created from the drawings enables the AMR to travel along the designated path automatically.

5.2. Methods

Figure 10(a) shows the AMR constructed by altering a wheelchair (WHILL model C: WHILL Inc.) to carry some pieces of small baggage. The PC runs ROS on Ubuntu 18.04, and Autoware is utilized for the autonomous driving control.

The experiment is conducted in the same building as mentioned in Sections 3 and 4. The map is selected as the one with the 0.4-m filter size from which the best localization result was obtained. In **Fig. 10(b)**, the desired path planned on the created map is depicted as a rightward curve. Following this path in the experiment, the AMR will turn to the right and go straight.

5.3. Results and Remarks

In all five runs, the AMR successfully travelled to the goal position following the desired path under the condition described in the previous section. The paths of the five experiments are over-drawn in **Fig. 10(c)**. **Fig. 10(d)** shows the photos taken during one of the experiments.

The expected movement is being achieved with the localization based on the map created from the drawings.

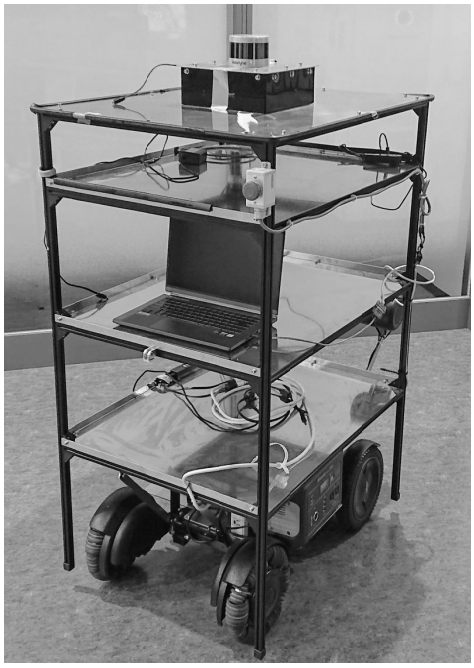
6. Conclusion

This paper proposed a map creation method for 3D LiDAR localization. The following two aspects are apparent: the map creation from the drawings, and the tablet scan of the non-existing objects on the map.

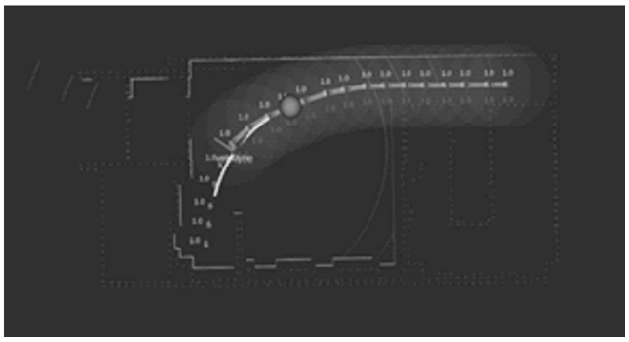
Regarding the first aspect, we focused on the fact that the place the AMR is supposed to run is nearly entirely inside of the buildings: there will be a drawing for the indoor environments. These drawings are usually drawn with the exact relative size as the actual building, though the scale is different. Thus, if the scale is adjusted at the final step, the map with the actual scale will be easily obtained. Because the map has a PCD format [e], we created the 3D PCD map file directly from the drawings by distributing the PCD at the position of the walls or partitions and accumulating them to the vertical direction.

The second aspect has an advantage in that the tablet computer is widely utilized in our daily life. This compact computer replaces the big AMR equipping the LiDAR in scanning the environment. The problem of appropriately placing the tablet scan data on the PCD map was solved by introducing the matching algorithm, NDT matching, which detects the most probable position of the scanned object using the simultaneously scanned reference data if it is initially placed close to the correct position.

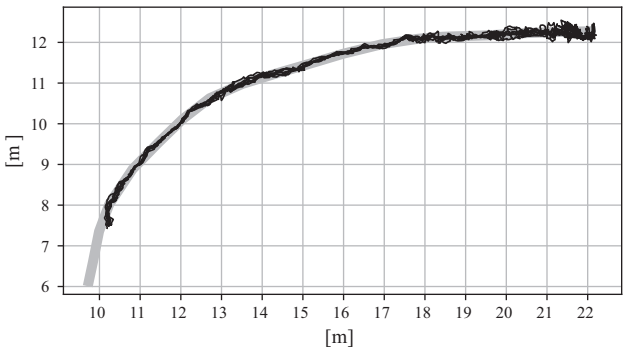
The first of three experiments were conducted to determine an important parameter: the size of the voxel grit filter, which determines the gap size along the wall surface on the PCD map. In this experiment, for the localization at the entrance of the building, the optimal size of the voxel grid filter was 0.4 m. This experiment also demonstrated the effectiveness of the tablet scan data. The tablet scan data enhance not only the matching level of the LiDAR scan data but also the accuracy of the estimated position. The second experiment showed that the automatic position correction of the tablet scan data worked as we intended. In this experiment, the translational deviation allowable in the worst direction toward the wall was 0.7 m,



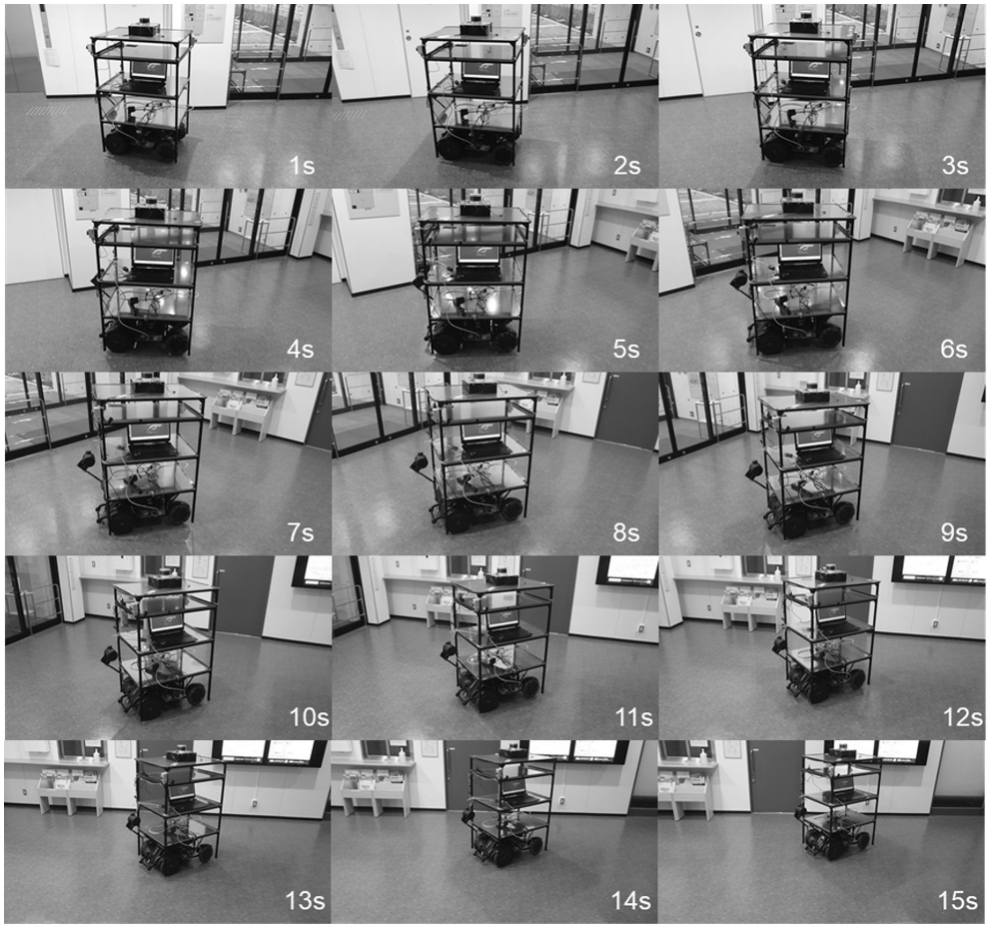
(a) AMR.



(b) The desired path on the map.



(c) The paths of five automatic driving experiments with desired path. The bold gray line is the desired path.



(d) Photos during the autonomous driving.

Fig. 10. Experiment of the autonomous driving using the map created from the drawing.

while the allowable rotational deviation was 10°. Regardless, the tablet scan data should be placed as close to its true position as possible. The third experiment succeeded in the autonomous driving of AMR without losing its position. This means that the map created using the proposed method can be available for the localization during autonomous driving.

Although we established a map creation method, the processes were not automatic, i.e., some manual tasks are needed, such as the elimination of the dimension lines and their numbers and the removal of the target object from the tablet scan data remaining the reference object data. We are working on a software program to reduce these manual tasks. Another problem is that some parameters, such as the size of the voxel grid filter, depend on the environment where the AMR travels. The airplane factory is huge, and thus, the optimal size of the voxel grid filter may differ from the one used in this paper. We believe this optimal size of the voxel grid filter might be related to the grid size of the NDT matching algorithm. It was set to 1 m in our experiments, while the optimal size of the voxel grid filter was 0.4 m. The voxel grid filter with a size several times less than the grid size of NDT matching algorithm should be selected from this result. Considering this prediction, we should create maps for various environmental sizes and compare the optimal parameter to find some tendencies through the experiments.

Acknowledgments

A part of this study is supported by "Human resource development and research project on production technology for aerospace industry: Subsidy from Gifu Prefecture."

References:

[1] M. Tomono, "Simultaneous Localization and Mapping," Ohmsha, 2018 (in Japanese).

[2] M. U. Khan, S. A. A. Zaidi, A. Ishtiaq, S. U. R. Bukhari, S. Samer, and A. Farman, "A comparative survey of LiDAR-SLAM and LiDAR based sensor technologies," 2021 Mohammad Ali Jinnah University Int. Conf. on Computing (MAJICC), pp. 1-8, 2021.

[3] M. Chghaf, S. Rodriguez, and A. E. Ouardi, "Camera, LiDAR and multi-modal SLAM systems for autonomous ground vehicles: a survey," J. of Intelligent & Robotic Systems, Vol.105, No.1, pp. 1-35, 2022.

[4] B. Schwarz, "Mapping the world in 3D," Nature Photonics, Vol.4, No.7, pp. 429-430, 2010.

[5] X. Gu, X. Wang, and Y. Guo, "A review of research on point cloud registration methods," IOP Conf. Series: Materials Science and Engineering, Vol.782, Article No.022070, IOP Publishing, 2020.

[6] P. Biber and W. Straßer, "The normal distributions transform: A new approach to laser scan matching," Proc. 2003 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS 2003), Vol.3, pp. 2743-2748, 2003.

[7] M. Magnusson, A. Lilienthal, and T. Duckett, "Scan registration for autonomous mining vehicles using 3D-NDT," J. of Field Robotics, Vol.24, No.10, pp. 803-827, 2007.

[8] T. Takubo, T. Kaminade, Y. Mae, K. Ohara, and T. Arai, "NDT scan matching method for high resolution grid map," 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 1517-1522, 2009.

[9] J. P. Saarinen, H. Andreasson, T. Stoyanov, and A. J. Lilienthal, "3D normal distributions transform occupancy maps: An efficient representation for mapping in dynamic environments," The Int. J. of Robotics Research, Vol.32, No.14, pp. 1627-1644, 2013.

[10] A. Yilmaz, E. Sumer, and H. Temeltas, "A precise scan matching based localization method for an autonomously guided vehicle in smart factories," Robotics and Computer-Integrated Manufacturing, Vol.75, Article No.102302, 2022.

[11] Z. Zhou, L. Li, A. Fürsterling, H. J. Durocher, J. Mouridsen, and X. Zhang, "Learning-based object detection and localization for a mobile robot manipulator in sme production," Robotics and Computer-Integrated Manufacturing, Vol.73, Article No.102229, 2022.

[12] M. Pinto, H. Sobreira, A. P. Moreira, H. Mendonca, and A. Matos, "Self-localisation of indoor mobile robots using multi-hypotheses and a matching algorithm," Mechatronics, Vol.23, No.6, pp. 727-737, 2013.

[13] S. Zhang, X. Tan, and Q. Wu, "Self-positioning for mobile robot indoor navigation based on wheel odometry, inertia measurement unit and ultra wideband," 2021 5th Int. Conf. on Vision, Image and Signal Processing (ICVISP), pp. 105-110, 2021.

[14] K. Takahashi, J. Arima, T. Hayata, Y. Nagai, N. Sugiura, R. Fukatsu, W. Yoshiuchi, and Y. Kuroda, "Development of edge-node map based navigation system without requirement of prior sensor data collection," J. Robot. Mechatron., Vol.32, No.6, pp. 1112-1120, 2020.

[15] S. Hoshino and H. Yagi, "Mobile robot localization using map based on cadastral data for autonomous navigation," J. Robot. Mechatron., Vol.34, No.1, pp. 111-120, 2022.

[16] R. Wang, J. Peethambaran, and D. Chen, "LiDAR point clouds to 3-D urban models: a review," IEEE J. of Selected Topics in Applied Earth Observations and Remote Sensing, Vol.11, No.2, pp. 606-627, 2018.

[17] S. Ochmann, R. Vock, R. Wessel, and R. Klein, "Automatic reconstruction of parametric building models from indoor point clouds," Computers & Graphics, Vol.54, pp. 94-103, 2016.

[18] J. R. Parent, C. Witharana, and M. Bradley, "Mapping building interiors with LiDAR: Classifying the point cloud with ArcGIS," Int. Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences, 2021.

[19] K. Babacan, L. Chen, and G. Sohn, "Semantic segmentation of indoor point clouds using convolutional neural network," ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences, Vol.4, pp. 101-108, 2017.

[20] Y. Wang, Z. Zhang, R. Zhong, L. Sun, S. Leng, and Q. Wang, "Densely connected graph convolutional network for joint semantic and instance segmentation of indoor point clouds," ISPRS J. of Photogrammetry and Remote Sensing, Vol.182, pp. 67-77, 2021.

[21] Y. Wu, Z. Yan, S. Cai, G. Li, Y. Yu, X. Han, and S. Cui, "Pointmatch: A consistency training framework for weakly supervisedsemantic segmentation of 3D point clouds," arXiv preprint arXiv:2202.10705, 2022.

[22] E. Takeuchi and T. Tsubouchi, "A 3-D scan matching using improved 3-D normal distributions transform for mobile robotic mapping," 2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 3068-3073, 2006.

[23] R. Kaneko, Y. Nakamura, R. Morita, and S. Ito, "PCD map creation from factory design drawing for LiDAR self-localization of autonomous mobile robot," Proc. of the Twenty-Seventh Int. Symposium on Artificial Life and Robotics 2022 (AROB 2022), pp. 569-574, 2022.

[24] Y. Kitsukawa, S. Kato, N. Akai, E. Takeuchi, and M. Edahiro, "Field testing of self-driving vehicles: lessons learned on localization," Int. Association of Traffic and Safety Science, Vol.42, No.2, pp. 48-53, 2017.

Supporting Online Materials:

[a] Voxel Grid filter. https://pointclouds.org/documentation/classpcl_1_1_voxel_grid_3_01pcl_1_1_p_c_l_point_cloud2_01_4.html [Accessed December 24, 2021]

[b] ROS Website. https://www.ros.org [Accessed May 27, 2022]

[c] Autoware developer Website. https://gitgub.com/CPFL/Autoware [Accessed May 15, 2020]

[d] How to use Normal Distributions Transform. https://pointclouds.org/documentation/tutorials/normal_distributions_transform.html [Accessed July 26, 2022]

[e] Point Cloud Library Website. https://pointclouds.org [Accessed July 26, 2022]

Name:
Satoshi Ito

ORCID:
0000-0003-2550-8153

Affiliation:
Faculty of Engineering, Gifu University

Address:
1-1 Yanagido, Gifu 501-1193, Japan

Brief Biographical History:
1991-1993 Graduate School of Engineering, Nagoya University
1994-1999 Bio-Mimetic Control Research Center, RIKEN
1999- Faculty of Engineering, Gifu University

Main Works:
- Robotics
- Control engineering
- Motor control and learning

Membership in Academic Societies:
- Institute of Electrical and Electronics Engineers (IEEE)
- The Robotics Society of Japan (RSJ)
- The Society of Instrument and Control Engineers (SICE)
- The Japan Society of Mechanical Engineers (JSME)

Name:
Ryutaro Kaneko

Affiliation:
Graduate School of Natural Science and Technology, Gifu University

Address:
1-1 Yanagido, Gifu 501-1193, Japan

Brief Biographical History:
2000-2022 Graduate School of Natural Science and Technology, Gifu University
2022- Panasonic Corporation

Main Works:
- Robotics

Membership in Academic Societies:
- The Robotics Society of Japan (RSJ)

Name:
Takumi Saito

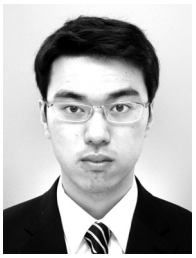
Affiliation:
Graduate School of Natural Science and Technology, Gifu University

Address:
1-1 Yanagido, Gifu 501-1193, Japan

Brief Biographical History:
2022- Graduate School of Natural Science and Technology, Gifu University

Main Works:
- Robotics

Membership in Academic Societies:
- The Robotics Society of Japan (RSJ)



Name:
Yuji Nakamura

Affiliation:
Faculty of Engineering, Gifu University

Address:
1-1 Yanagido, Gifu 501-1193, Japan

Brief Biographical History:
2022- Mitsui E&S Machinery Co., Ltd.

Main Works:
- Robotics