https://doi.org/10.20965/jrm.2019.p0180

Paper:

Navigation Based on Metric Route Information in Places Where the Mobile Robot Visits for the First Time

Asahi Handa, Azumi Suzuki, Hisashi Date, Ryohsuke Mitsudome, Takashi Tsubouchi, and Akihisa Ohya

Graduate School of Systems and Information Engineering, University of Tsukuba
1-1-1 Tennodai Tsukuba-shi, Ibaraki 305-8573, Japan
E-mail: handa-a@roboken.iit.tsukuba.ac.jp, azumi.suzuki@tier4.jp, {hdate@iit, mitsudome-r@roboken.iit, tsubo@iit, ohya@cs}.tsukuba.ac.jp
[Received November 1, 2018; accepted March 6, 2019]

In this study, we propose a navigation system that guides a robot at a location visited for the first time, without developing a map in advance. First, it estimates the position of a path that exists on the local map by matching the metric route information and the local map generated by simultaneous localization and mapping (SLAM); this is achieved by using a particle filter. Then, the robot travels to the destination along the estimated route. In this system, the geometric accuracy of the route information specified in advance and the accuracy of the map generated by SLAM are essential. Furthermore, it is necessary to recognize the traversable area. The experiment performed verifies the matching of the route information and local map. In the autonomous running experiment, we conduct a trial run on a course set up at the University of Tsukuba.

Keywords: mobile robot, route information, navigation, traversability analysis

1. Introduction

A mobile robot must be capable of autonomously navigating from its current position to its destination for it to carry out activities in the real world. Autonomous navigation from an arbitrary initial position to the destination is generally based on localization by matching the point cloud data obtained by Light Detection and Ranging (LIDAR) and map data created in advance. This method has been used by most robots that have successfully completed the autonomous navigation task at the Tsukuba Challenge [a], a Real World Robotics Challenge for autonomous navigation robots. It is necessary in this method to visit and manually run the robot at the planned site beforehand in order to gather data required to create the map. Substantial effort and time are required for the map to cover an extensive area; moreover, this method cannot be used to immediately operate the robot at a location where it has never been previously. However, humans are capable of arriving at the destination using an approximately sketched map; this is because they have a refined recognition capability that enables them to match landmarks on the map with the real world. Nevertheless, it has been challenging in the past to enable robots to autonomously navigate using similar recognition capabilities.

In this study, we propose a method of matching metric route information and a highly accurate map produced by Simultaneous Localization and Mapping (SLAM); it is aimed at constructing a navigation system by which the robot can navigate to the destination even in locations encountered for the first time. The route information is produced by manually tracing the desired route; the route is assumed to be traversable based on aerial photographs captured in advance, or architectural plans. Meanwhile, the robot recognizes the traversable areas in its surroundings and creates a map of the traversable areas by recording them on the map created by SLAM. By overlaying the route information on it, the correspondence of the map to the route information is obtained. A feature of the proposed method is the precondition that the route information be geometrically accurate; this enables the robot to autonomously navigate as long as it is capable of recognizing the traversable areas without requiring a high object recognition capability as in humans.

From the above perspective, the authors have been engaged in research on methods by which a robot can navigate locations that it encounters for the first time [1–4]. In reference [1], we verified the validity of the route localization function of the proposed method in an indoor environment. In references [2–4], the route localization function was further extended and implemented with a method to recognize traversable areas based on three-dimensional LIDAR so as to enable the robot to operate in outdoor environments. Trial runs were then carried out in real environments such as a university grounds and the Tsukuba Challenge site.

In this paper, we provide an overview of the proposed method and past experiments, and report our observations verified through the experiments; the observations include the conditions under which the proposed method is valid, types of environments suitable for recognition outdoors, and recognition capability required by the robot for it to safely engage in activities in real environments. The paper is organized as follows: Section 2 reviews the issues unresolved in related studies and describes the approach adopted by this study to address those issues. Section 3 presents the theory underlying the proposed method. Section 4 describes the implementation of the proposed method. Section 5 presents the results of the autonomous navigation experiment conducted on the grounds of the University of Tsukuba; the section also presents the discussions. Section 6 enumerates our conclusions.

2. Related Studies

The topological approach is similar to the manner in which humans interpret maps [5]. Representing the map as a graphical structure, an estimate of the present location is obtained from characteristic landmarks in the environment. Because of its low accuracy and the need to define target landmarks in advance, this approach has not been used for navigation in locations encountered for the first time [6].

The metric approach is diametrically opposed to the topological approach. Localization based on the matching of a map prepared in advance and sensor data, as described above, belongs to this category. There are also studies that lie in between these two approaches, wherein the information provided in advance is minimized and the localization is carried out with high accuracy.

Reference [7] employed the information of building areas extracted manually from aerial photos and matches it with measured 3D environmental data to carry out localization in an unspecified environment. This method is limited to environments with buildings in the vicinity that can be visually recognized in aerial photographs; moreover, it entails a high manpower cost in advance because the map must be constructed manually. Reference [8] employed two-dimensional street maps and achieved localization using the squared-loss mutual information as an index. It indicated that errors can arise in cases in which a street displayed in the 2D street map exhibits a topology considerably different from reality; however, the authors did not discuss how this can affect autonomous navigation. Furthermore, at the time of its publication, the boundary information had to be manually produced in the map. In reference [9], localization was achieved by directly using the data of OpenStreetMap, although this was not applied to autonomous navigation; meanwhile, localization was limited to indicating the approximate area of a specified street being navigated by the robot. Reference [10] also employed OpenStreetMap data. The moving robot was localized on OpenStreetMap using GPS position information; moreover, autonomous navigation was conducted by using LIDAR to detect the road surface. However, the experiments were conducted with simple routes, and the method could not address road forks and other topologies. In references [11, 12], autonomous navigation systems were constructed from route information produced from aerial photographs, road-following navigation, intersection recognition, and azimuth detection. Whereas autonomous navigation was achieved in actual urban settings, the attached information such as intersections had to be added manually to the route information. Reference [13] used an edge-node graph produced from an electronic map such as Google Map; moreover, the edge of the edge-node graph corresponding to the position of the mobile robot was estimated by odometry. A simple electronic map may not necessarily coincide geometrically with the actual environment; furthermore, localization errors are likely to occur when similar routes exist or owing to odometry error.

A car navigation system [14] estimates self-position by matching the travelled trajectory generated from position information obtained by dead reckoning against an electronic map. Because the human driver recognizes the road while driving, no sensing of the real environment is carried out; moreover, localization is based on the assumption that the vehicle is following the road.

In reference [15], particle filters were used to estimate the positions of manually set waypoints; furthermore, the method prevented the waypoints from being set on untraversable areas. The waypoints were estimated and corrected based on the traversable and untraversable areas detected by LIDAR and a stereo camera; this enabled the robot to navigate even when the environmental map or route information was indistinct. However, the waypoints were likely to have been adjusted erroneously at branches because they were individually adjusted; this issue is not addressed in reference [15]. Furthermore, the authors indicated the issue of erroneous adjustments when a moving object, such as a pedestrian, was present at a waypoint position.

Because the method proposed by the present study employs route information based on aerial photographs, no advance measurements are necessary, similar to references [7, 8]. The major difference lies in the provision of only the geometric information of the specified route, without the need to attach additional information on environmental features by extracting the road boundaries or building shapes from the aerial photos. Furthermore, unlike reference [13], the metric route information is overlaid on the traversable areas; this enables the localization of the route within the recognized environment. The car navigation system [14] matches the traveled trajectory against the map based on the assumption that the human driver is keeping the vehicle on the road; therefore, it is similar to the present study in that localization is carried out metrically. From this perspective, the objective of the present study can be considered as the realization of a completely autonomous system by using sensors to automate the driver's recognition and driving. Because the recognition of traversable areas (which the proposed method requires) is equivalent to the recognition of obstacles (necessary to achieve the robot's autonomous navigation), we consider that refining this function will enhance the overall system stability.

3. Route Information and SLAM-Based Route Localization

3.1. Outline of Proposed Method

In this study, we employ metric route information created manually from geometrically accurate drawings such as architectural plans and aerial photographs. We use the term metric from the perspective that the captured or displayed object's actual configuration and relative scale are preserved. Because the map produced by SLAM is also likely to be metric, the localization necessary for navigation can be obtained by matching the two. If we assume that the route specified by the human operator and the map produced by SLAM both recreate the topology and shapes of roads and objects without error, the corresponding shapes should match substantially. In this case, the route localization in the proposed method and the determination of the robot's initial position and attitude in the real world are equivalent problems.

However, in reality, map distortion owing to errors of SLAM and errors in the manually-produced route exist. The likely factors of route errors include distortions in the aerial photos and distance errors arising when the selected route is in an environment containing altitude difference. Nevertheless, we assume them to be negligible in this study. Meanwhile, distortions in the map produced by SLAM cannot be prevented and therefore, must be accounted for by the system.

We outline the proposed system below, and in the following sections, describe the method of localization and measures to address the likely issues in detail:

- The route information by which the robot navigates through the traversable area is manually produced from geometrically accurate aerial photos and other sources.
- The robot produces the local map using SLAM as it navigates and detects the traversable areas. The areas where temporary obstacles such as pedestrians are present are assessed as traversable.
- A particle filter is used to estimate the route on the traversable area map, and the robot's self-position is obtained by coordinate transformation.
- The robot navigates based on the estimated route position and self-position.

3.2. Creation of Route Information

Aerial photos available on Google Earth or other websites are used to create the route information. However, note that a detailed architectural plan, if available, may rather be used when navigating indoors; this is because it suffices for the route information to be geometrically accurate. The route information consists of the coordinates of the waypoints after a two-dimensional coordinate system Σ_W has been superposed on the aerial map or plan; the information is provided in the form of an adjacency list with the waypoints as the vertices.

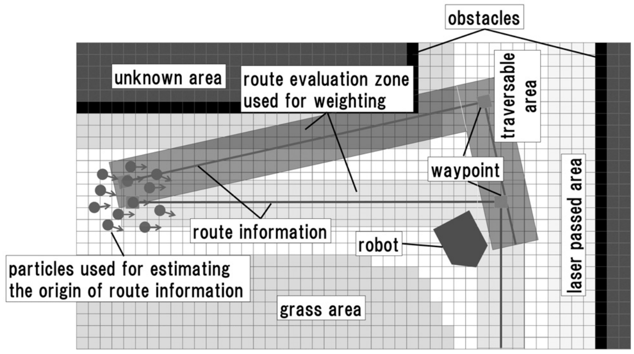


Fig. 1. Conceptual diagram of proposed method: estimate the origin of the route information in the traversable area map. The route information with particles as the origin is superimposed on the traversable area map. The route information is specified as an area with a certain width, and the particles are weighted according to the ratio of the traversable area included. The weight of the particles in which the traversable area is large in the route evaluation zone is high (light-gray route evaluation zone). The weight of particles with a small number of traversable areas included in the route evaluation zone is low (dark-gray route evaluation zone).

3.3. Detection of Traversable Areas and Map Creation

The traversable area is detected from the point cloud data obtained by three-dimensional LIDAR. The paved road surfaces are mainly detected as traversable areas; this is because it is feasible to detect the difference in elevation created by steps on the road surface or distinguish between paved road surfaces and grassy areas from the point cloud data. As shown in **Fig. 1**, the traversable areas are recognized by distinguishing the paved road surfaces from the steps and grassy areas. This route estimation method can be used for traversable areas detected by other methods. The specific implementation is described in Section 4.

In addition, a local map is produced by SLAM and used to estimate the route position, which is described below, and the recognized traversable areas are accumulated and stored on an occupancy grid map. The two-dimensional coordinate system of the map produced in this manner is denoted by Σ_M.

3.4. Localization Based on Route Estimation

Because the route information is specified by coordinates in the Σ_W coordinate system (defined in Section 3.2), the self-position and attitude in this coordinate system is necessary for navigation. Meanwhile, the robot determines its self-position and attitude in the Σ_M coordinate system using SLAM, as defined in Section 3.3; therefore, the coordinate transformation $T_{M\to W}$ from Σ_M to Σ_W is necessary to obtain the robot's self-position and attitude in the Σ_W coordinate system. This $T_{M\to W}$ coordinate transformation is estimated using a particle filter.

Figure 1 shows a conceptual diagram of the proposed method. The procedure to estimate the route position using the particle filter and the method of determining the self-position from coordinate transformation are described below. The particle filter used for the route estimation is used exclusively for this purpose and is unrelated to the particle filter in SLAM.

3.4.1. Route Estimation Using Particle Filter

The coordinate transformation $T_{M \to W}$ from Σ_M to Σ_W is determined by the coordinates of the origin of the Σ_W coordinate system in Σ_M and relative attitude of the two coordinate systems. Therefore, we consider the particle $\boldsymbol{x}_k^{[m]} = \left(x_k^{[m]}, y_k^{[m]}, \theta_k^{[m]}\right)$ as the hypothesis, and its weight $w_k^{[m]}$. Here, k denotes the discrete time, and m is the particle index. Because the coordinate transformation $T_{M \to W}$ is essentially static, the prediction model in the particle filter is assumed to be stationary.

Step 1: Initialization of particle filter

The particles are assumed to exhibit a Gaussian distribution with covariance matrix $\Sigma = \mathrm{diag}\left(\sigma_x^2, \sigma_y^2, \sigma_\theta^2\right)$ centered at the robot's initial position and attitude $\boldsymbol{x}_{init} = (x_{init}, y_{init}, \theta_{init})$. There are N particles, and the initial weight of each particle is $1/N$. The initialization is expressed as follows:

$$\boldsymbol{x}_0^{[m]} \sim \mathcal{N}(\boldsymbol{x}_{init}, \Sigma) \quad \ldots \ldots \ldots \quad (1)$$

$$w_0^{[m]} = \frac{1}{N} \quad \ldots \ldots \ldots \quad (2)$$

Step 2: Weighting

The particles are evaluated based on the extent of traversable area included in the route evaluation zone when the route information translated and rotated by $x_k^{[m]}, y_k^{[m]}, \theta_k^{[m]}$ is overlaid on the traversable area map. Because the estimated route must be navigable by the mobile robot, a route with a width approximately equal to the robot's body is used for evaluation. The robot scans areas where this route overlaps the traversable area map and counts the total number of grids in each area type. The number of grids in the respective areas is determined by observing the traversable area map, z_k, and particle $\boldsymbol{x}_k^{[m]}$. The numbers of grids in the traversable, obstacle, unknown, and grassy areas, and the area over which the horizontal laser passes unobstructed are denoted by $N_{tr}^{[m]}$, $N_{ob}^{[m]}$, $N_{un}^{[m]}$, $N_{gr}^{[m]}$, and $N_{fr}^{[m]}$, respectively; these are multiplied by coefficients C_{tr}, C_{ob}, C_{un}, C_{gr}, and C_{fr}, respectively, and then summed to obtain $W^{[m]}$, which represents the likelihood of the particle. This is expressed as follows:

$$W^{[m]} = C_{tr}N_{tr}^{[m]} + C_{ob}N_{ob}^{[m]} + C_{un}N_{un}^{[m]} + C_{gr}N_{gr}^{[m]} + C_{fr}N_{fr}^{[m]} \quad \ldots \ldots \ldots \quad (3)$$

Step 3: Updating of particle filter and computation of estimated value

Estimation is carried out using the posterior probability $p(\boldsymbol{x}_k \mid z_k)$, prior probability $p(\boldsymbol{x}_k)$, and likelihood $p(z_k \mid \boldsymbol{x}_k)$. From Bayesian inference, the posterior probability is expressed as

$$p(\boldsymbol{x}_k \mid z_k) \propto p(z_k \mid \boldsymbol{x}_k)\, p(\boldsymbol{x}_k) \quad \ldots \ldots \quad (4)$$

Because the prediction model is assumed to be static in the proposed method, the prior probability $p(\boldsymbol{x}_k)$ is equal to the distribution a time-step before, $p(\boldsymbol{x}_{k-1})$, and its sampling value $p\left(\boldsymbol{x}_{k-1}^{[m]}\right)$ is equal to the particle weight $w_{k-1}^{[m]}$. The likelihood $p(z_k \mid \boldsymbol{x}_k)$ is given by $W^{[m]}$. Assuming that the posterior distribution is obtained as the updated particle weight $w_k^{[m]}$, the continuous distribution of Eq. (4) can be expressed as a sample approximation as follows:

$$w_{k-1}^{[m]} = p\left(\boldsymbol{x}_{k-1}^{[m]}\right) \sim p(\boldsymbol{x}_k) \quad \ldots \ldots \quad (5)$$

$$W^{[m]} p\left(z_k \mid \boldsymbol{x}_k^{[m]}\right) \sim p(z_k \mid \boldsymbol{x}_k) \quad \ldots \ldots \quad (6)$$

$$w_k^{[m]} = p\left(\boldsymbol{x}_k^{[m]} \mid z_k\right) \sim p(\boldsymbol{x}_k \mid z_k) \quad \ldots \ldots \quad (7)$$

Thus, we obtain the update rule for the particle weights as

$$w_k^{[m]} = W^{[m]} w_{k-1}^{[m]}. \quad \ldots \ldots \ldots \quad (8)$$

The weighted average $\hat{\boldsymbol{x}}_k$ of all particles represents the estimated coordinates and attitude of the origin of the Σ_W coordinate system in the Σ_M coordinate system.

$$\hat{\boldsymbol{x}}_k = \left(\hat{x}_k, \hat{y}_k, \hat{\theta}_k\right) = \frac{\sum_{m=1}^{N} w_k^{[m]} \boldsymbol{x}_k^{[m]}}{\sum_{m=1}^{N} w_k^{[m]}} \quad \ldots \ldots \quad (9)$$

Step 4: Resampling

A specified number of the low-weight particles are selected and re-scattered in the vicinity of the high-weight particles. In the present case, 5% of the total particles are selected from the lowest-weight particles and scattered in the vicinity of the high-weight particles.

3.4.2. Acquiring Self-Position by Coordinate Transformation

Denoting the robot's self-position and attitude in the Σ_M coordinate system by $(x_r^M, y_r^M, \theta_r^M)$, its self-position and attitude in the Σ_W coordinate system, $(x_r^W, y_r^W, \theta_r^W)$,

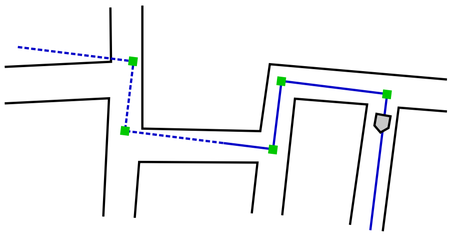


Fig. 2. Limited measurement criterion set to address map distortion. Because the dashed lines collide with the wall in the distorted map, they are not counted for route pose estimation.

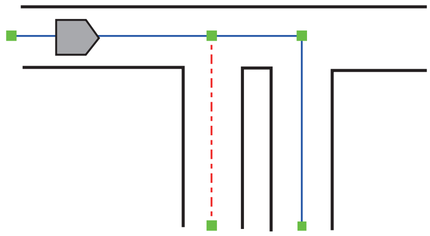


Fig. 3. Route information with branches. Although the dash-dot line is not the actual route, we evaluate whether it is on a traversable area, for route pose estimation.

is specified by the estimated value, $\hat{\boldsymbol{x}}_k = (\hat{x}_k, \hat{y}_k, \hat{\theta}_k)$ (obtained from the particle filter in the previous section), from the following coordinate transformation $T_{M \to W}$:

$$\begin{pmatrix} x_r^W \\ y_r^W \end{pmatrix} = \begin{pmatrix} \cos\hat{\theta}_k & \sin\hat{\theta}_k \\ -\sin\hat{\theta}_k & \cos\hat{\theta}_k \end{pmatrix} \begin{pmatrix} x_r^M - \hat{x}_k \\ y_r^M - \hat{y}_k \end{pmatrix} \quad (10)$$

$$\theta_r^W = \theta_r^M - \hat{\theta}_k \quad (11)$$

3.5. Methods to Address Distortion of Produced Map

Distortions are increasingly introduced in the SLAM-produced map as the robot travels longer distances. Thus, situations involving challenges in estimating the route position are likely when the proposed method is used to navigate long distances.

This is addressed by limiting the route section to be weighted to a range within which the map distortion stays below a permissible range. Specifically, the section of the route that lies within a certain distance from the base of the perpendicular drawn from the robot's self-position in the Σ_W coordinate system to the closest route, is used to calculate weights. In this manner, we expect that the route will be estimated correctly in the vicinity of the robot's self-position even when the entire route does not match (**Fig. 2**); this enables navigation along the route.

Meanwhile, when a straight or arc-following route extends over a long distance, localization in the forward direction can become uncertain. In such cases, the evaluated section to the rear is extended to include a corner.

3.6. Measures Against Branches

The localization accuracy of the proposed method is affected by the width and topology of the road. The position accuracy reduces after a wide road or a long straight section has been navigated. When a narrow branched section immediately follows such sections, the correct path is not likely to be selected because of the position error. When there are branches such as those shown in **Fig. 3**, the specified route information matches either branch as they are challenging to distinguish.

To address similar situations, the branch denoted by the dash-dot line in **Fig. 3** is attached in advance to the route information. This branch is evaluated in a similar manner

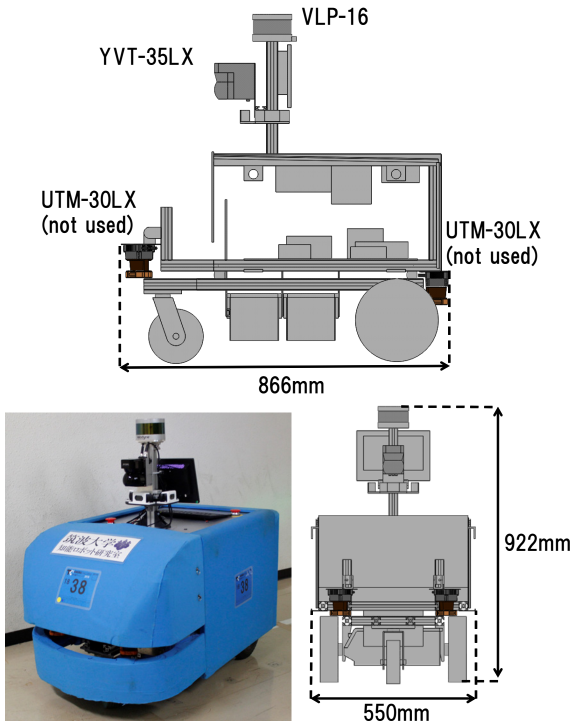


Fig. 4. Appearance of mobile robot.

as the route; if it lies in the traversable area, the weights of the corresponding particles are increased. This is expected to improve the accuracy.

4. The Applied Robot and Implementation of Proposed Method

The proposed method is implemented on the ROS[1] platform. The mobile robot shown in **Fig. 4** is used to carry out the autonomous navigation experiment described in the following sections. The robot employs a wheel encoder and IMU: NAV420 (Crossbow Technology Inc.) as internal sensors. As external sensors, the robot is equipped with two 3D LIDARs, namely, VLP-16 (Velodyne LIDAR, Inc.) and YVT-35LX (Hokuyo Automatic

1. Robot Operation System. http://www.ros.org/ [Accessed October 22, 2018]

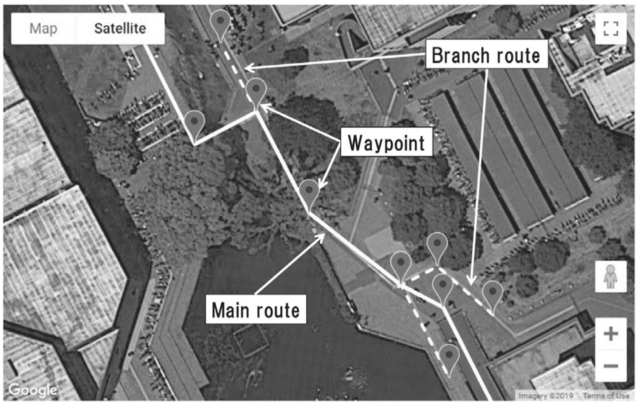


Fig. 5. Example of route information using Google maps API.

Co. Ltd.). The latter is for detecting steps and low lying obstacles at a close range in front of the robot, which are blind areas of VLP-16.

The functions required for autonomous navigation by the proposed method are implemented as follows.

4.1. Creation of Route Information

The robot is implemented with a web application that acquires the longitudinal and latitudinal coordinates of a point clicked on an aerial photo or street map on Google maps API and stores it as the route information. Because the present method is based on metric (i.e., geometrically accurate) information, it is unaffected by inaccuracies in the magnitudes of longitudinal/latitudinal coordinates as long as the relative geometric positions of the points forming the route are accurate.

An example of the produced route information is shown in **Fig. 5**. The waypoints are indicated by the pin icons, and the solid polyline connecting those points represents the route. The dashed polylines represent branches, as described earlier.

4.2. SLAM and Creation of Traversable Area Map

The slam_gmapping package of ROS is used as SLAM. It is a SLAM approach that employs Rao-Blackwellized particle filters to create grid maps [16]. Because it employs 2D laser scan data, the range of the laser beam reflected at the most-distant point in each line obtained with VLP-16 is used as the scan data. The slam_gmapping package produces a map; however, because we are mainly interested in distinguishing grassy areas, the traversable area map is constructed from the SLAM-acquired self-position based on the results of the recognition method described below.

4.3. Recognition of Traversable Area

VLP-16 is used to detect flat, grassless road surfaces. The velodyne_height_map package of ROS is used to detect flat road surfaces. In this package, the acquired 3D point cloud is divided into plane grids; any grid is assessed as flat if the difference between the maximum and minimum z-coordinates of the point cloud in the grid is sufficiently small. Any grid that is not assessed as flat is considered as an obstacle area.

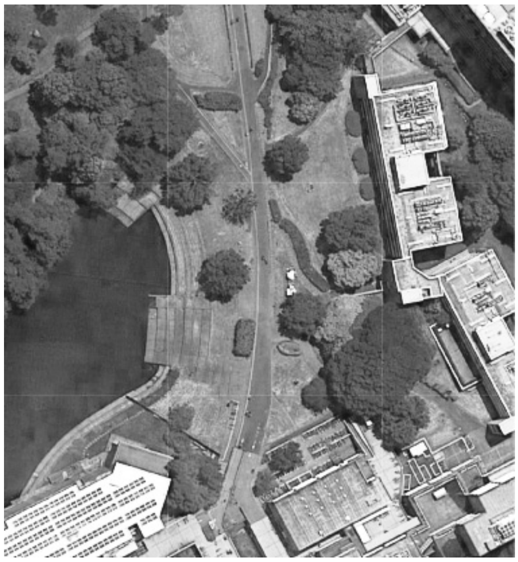


Fig. 6. Aerial photo of example environment.

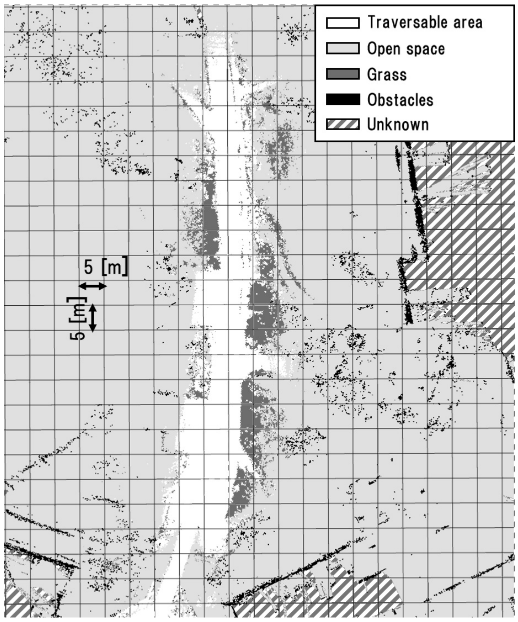
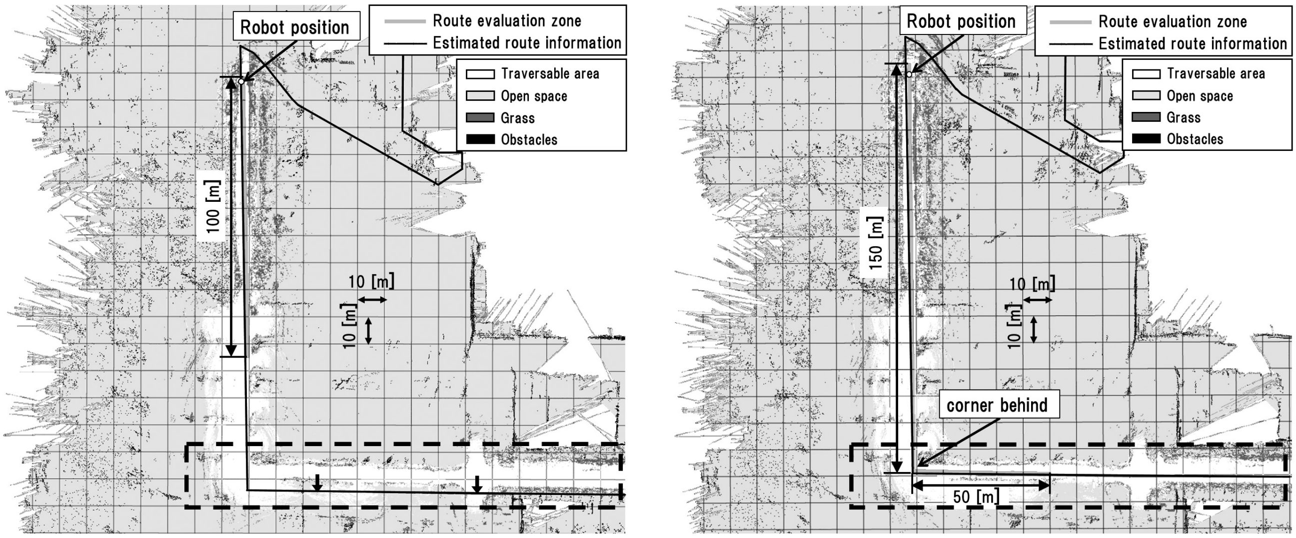


Fig. 7. Recognition result of example environment.

To detect grass in any area assessed as a flat road surface, we use a reported method based on the reflected remission of LIDAR [17, 18]. Specifically, we obtain the average remission value within each grid for point clouds in grids that are assessed as flat road surfaces using the velodyne_height_map; then, we use a threshold that depends on the distance from the sensor, to determine whether it is grass.

For the outdoor environment, an aerial photo of which is shown in **Fig. 6**, the produced traversable area map is shown in **Fig. 7**. The grids are color-coded as follows: white indicates traversable areas, black the obstacle areas,



(1) The case where interval for weighting is constant distance.

(2) The case where interval for weighting includes the corner behind.

Fig. 8. Difference in result of estimated route information by interval for weighting. (1) In the range surrounded by the dashed line, because an error occurs in the route estimation results, the route information deviates from the traversable area (represented by an arrow). (2) In the range surrounded by the dashed line, the route information is correctly estimated in the traversable area.

light-gray the free areas (over which the 2D laser scan beam, described in Section 4.2, passed without obstruction), dark-gray the grassy areas, and hatched pattern the unknown areas.

4.4. Navigation

Navigation is implemented with the ROS move base package; the target points consist of waypoints that have been coordinate-transformed to the Σ_M coordinate system based on the estimation results of the proposed method; moreover, the Dynamic Window Approach (DWA) is used to generate the trajectory [19].

DWA requires an obstacle map represented as a grid map to generate the trajectory. In our case, we employ the obstacle areas and grassy areas obtained when constructing the traversable area map, as well as the obstacle/flat area information obtained by applying the velodyne_height_map package to point clouds acquired by VLP-16 and YVT-35LX.

4.5. Route Evaluation Zone that Includes Corners to the Rear

As mentioned in Section 3.5, navigation-direction errors can occur on long, straight road sections. This is addressed by extending the route evaluation zone to the rear so as to include a corner. **Fig. 8** shows the results of the route estimation when the corner to the rear is not included and those when the route evaluation zone is extended to include the corner. In the former case, the route evaluation zone extends 100 m to the rear from the robot's self-position. In the latter case, when there are three adjacent points that form an angle (route shape) of less than 135° within a distance of 150 m to the rear, the route evaluation zone is extended to a point 50 m further back from the corner. A comparison of the results reveals that the route estimation results in a deviation of approximately 10 m in the longitudinal direction when the rear corner is not included; meanwhile, this deviation is suppressed when the route evaluation zone is extended to include the rear corner. Thus, we verify that extending the route evaluation zone to include the rear corner is a valid measure.

5. Autonomous Navigation Experiment on the Grounds of University of Tsukuba

5.1. Objective and Method of Experiment

To verify the validity of the proposed method, a route of distance approximately 2 km is set within the grounds of the University of Tsukuba, and an experiment is conducted to verify whether the robot is capable of autonomously navigating the entire route.

The following items were selected for evaluation because they are considered to be important factors affecting the success of the proposed method:

(1) Localization error.

(2) Effect of SLAM accuracy.

(3) Effect of environment recognition capability.

(4) Length of route evaluation zone.

Because the evaluation of item (1), which is to examine the localization performance of the proposed method, requires the manual matching of the SLAM-produced map and actual trajectory, it would require considerable time and effort to be carried out for the entire route. Therefore, it is evaluated for a few selected points. Specifically, we

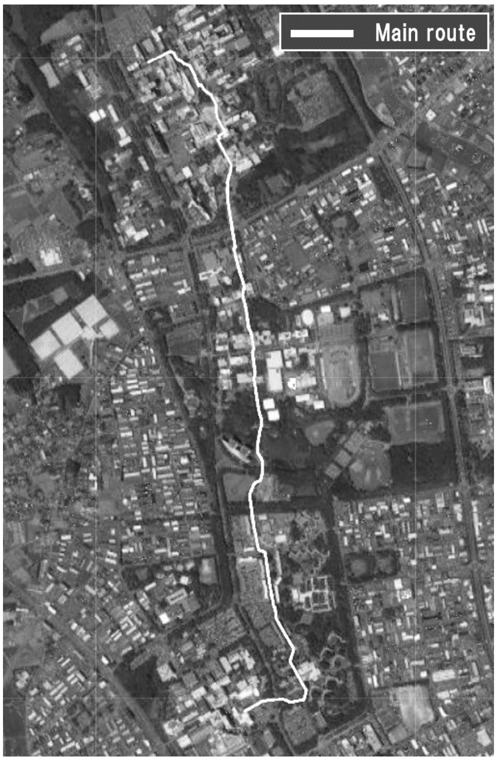


Fig. 9. Route information in University of Tsukuba.

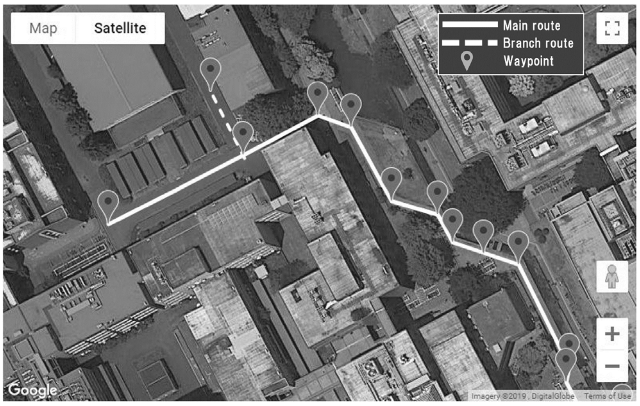


Fig. 10. Route information around the start position in University of Tsukuba.

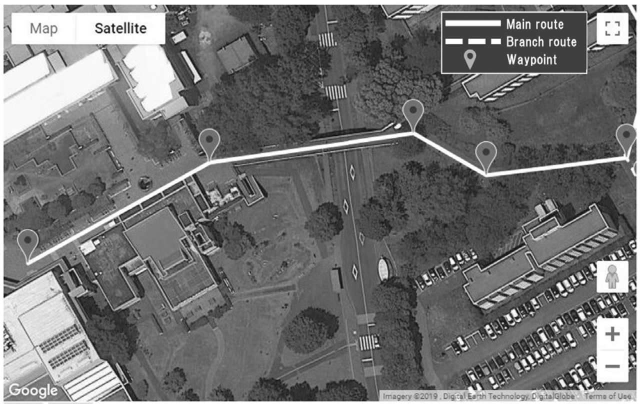


Fig. 11. Route information around the goal position in University of Tsukuba.

assess the route position error between the estimated route and the one prepared in advance at sections where the particles of the route-estimation particle filter are widely dispersed; this is likely to result in low localization accuracy.

The purpose of item (2) is to examine the impact of the localization accuracy of SLAM on the route estimation. The localization accuracy of SLAM is evaluated by examining its discrepancy against odometry [20]. We verify the effect that a large error of this type has on autonomous navigation.

Item (3) has been verified at a fundamental level in Section 4.3. In the present experiment, we examine whether an insufficient environment recognition capability can hinder autonomous navigation.

With regard to item (4), Section 4.5 describes the validity of including the corner to the rear in the evaluation zone. We examine its effect on autonomous navigation in the experiment.

The starting point of the route is set in front of Building L in Area 3 of the University of Tsukuba. The route then passes through the pedestrian deck and ends in front of the Medical Library in the Medical Area, a distance of approximately 2 km (**Fig. 9**). Magnified views of the starting and goal points are shown in **Figs. 10** and **11**, respectively.

The coefficients of Eq. (3) used to evaluate the likelihood of the route position estimation are presented in **Table 1**. The coefficients are determined heuristically so that the traversable areas are rated high whereas the other areas, in particular the grassy areas and obstacle areas, are rated low.

Table 1. Coefficients in Eq. (3) for tested system.

C_{tr}	C_{ob}	C_{un}	C_{gr}	C_{fr}
10	-2	-0.2	-1	-0.2

5.2. Experiment Results

Although there are points at which the robot comes to an emergency stop or its position has to be corrected manually, it is capable of autonomously navigating the route to the goal point without failure of route position estimation. There are nine spots where it is unable to continue autonomous navigation; images of those spots are shown in the order of occurrence in **Fig. 12**.

Figure 13 shows the standard deviations of the particles used for route estimation. It is apparent that the standard deviation peaks at four points: the 200 m, 800 m, 1,200 m, and 1,700 m points. To examine item (1), the errors between the route information provided in advance and the estimated route are determined at these four points. To compute the error, satellite photos used to create the advance route information are overlaid on the corresponding SLAM-generated maps for comparison with the estimated routes. **Fig. 14** shows the route information provided in advance and the estimated route positions at the four

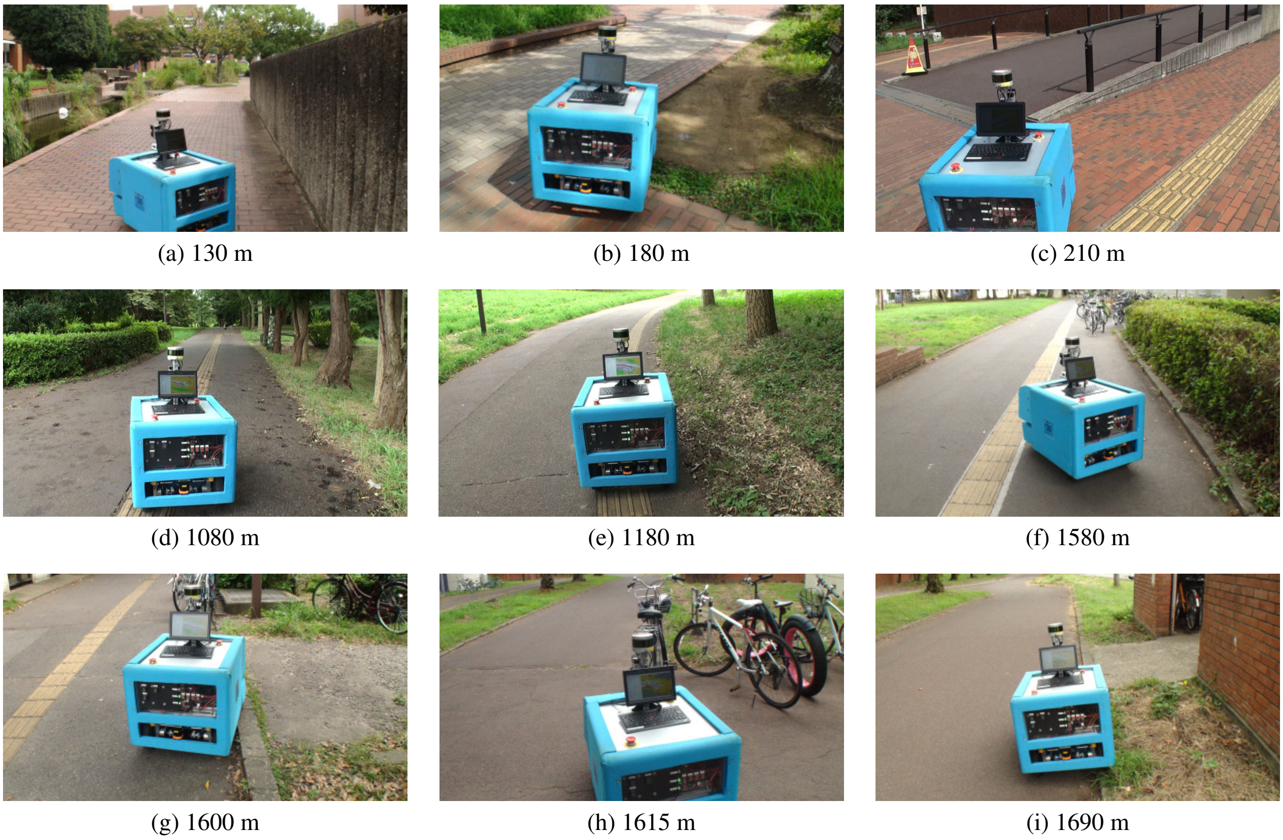


Fig. 12. Stacked points of the autonomous running experiment in university.

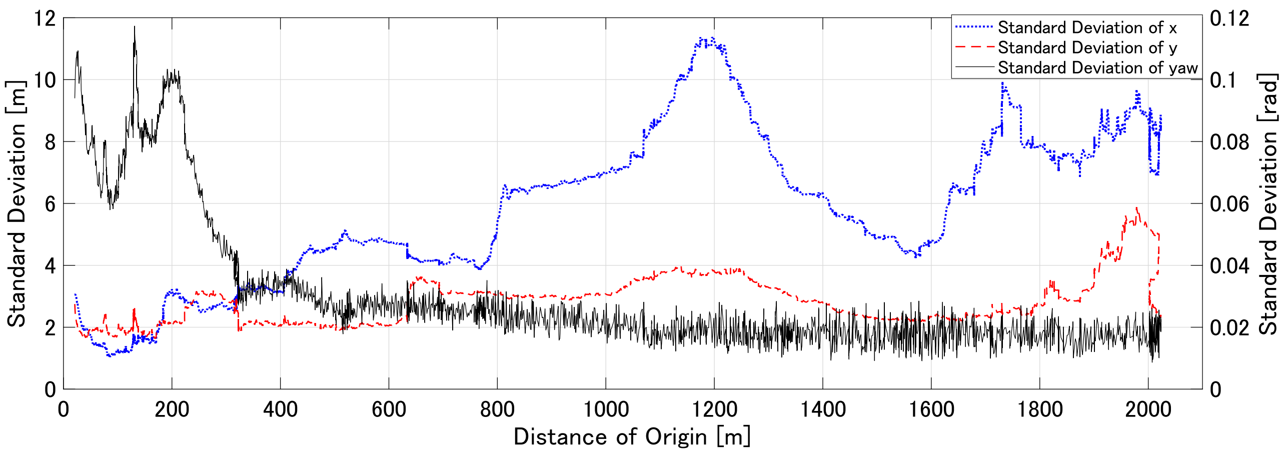
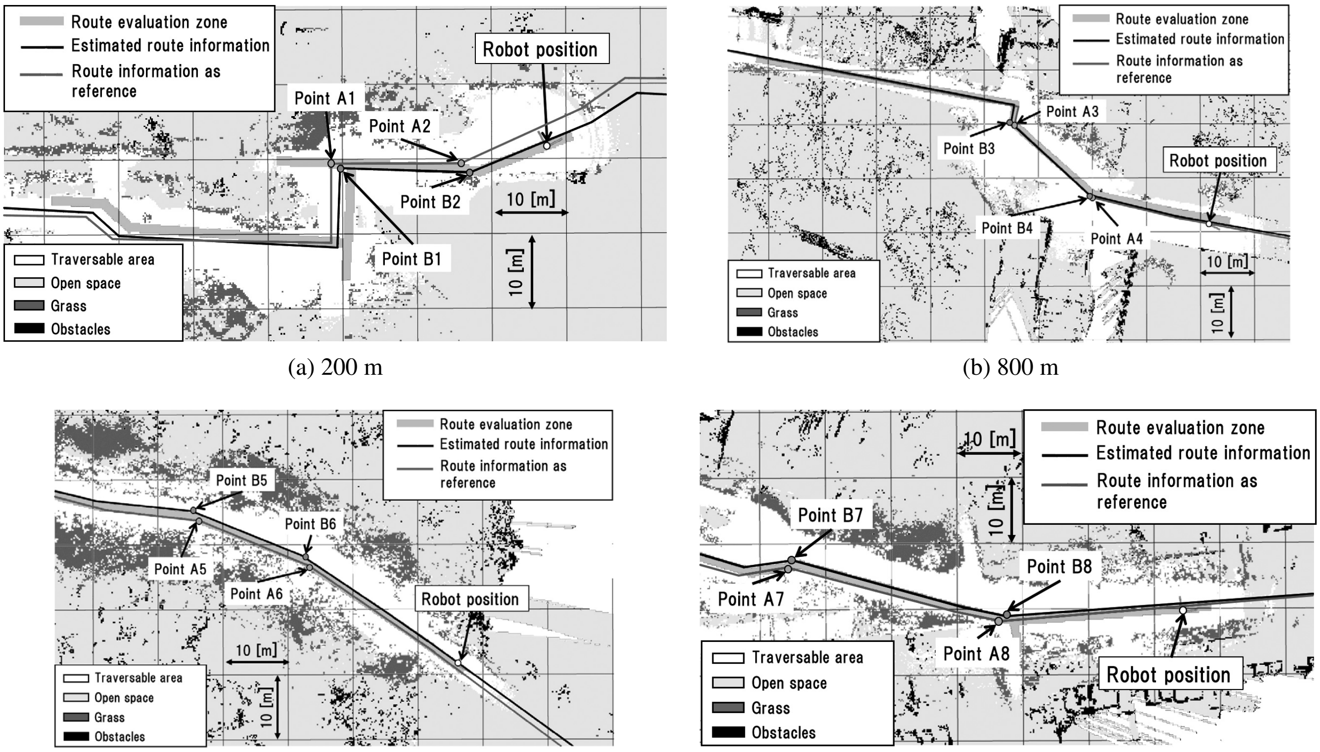


Fig. 13. Standard deviation of particles based on route position.

points; **Table 2** presents the distance errors between the two. It is apparent that the error is at most approximately 2 m; this indicates that the route is estimated within a range that does not depart from the path that should be adopted.

With respect to item (2), the accuracy of SLAM-based localization is evaluated in **Fig. 15**. It is based on the method of evaluating the dispersion of localization with reference to odometry (described in reference [20]); moreover, it plots the standard deviations of multiple odometry initialized every 1 m using the SLAM localization results. Large values indicate large discrepancies with odometry, which in turn indicates that the SLAM-produced map has become distorted. It is apparent that large discrepancies occurred between SLAM localization and odometry in the vicinities of 650 m, 1,600 m, and 1,950 m from the origin. Of these, the increased discrepancies at 650 m and 1,600 m do not correspond to any of the points shown in **Fig. 12**; thus, they had no effect in terms of continuation of autonomous navigation. However, the high discrepancy near the 1,600 m point is likely a factor in the disruption of the robot's autonomous navi-



(a) 200 m (b) 800 m

(c) 1200 m (d) 1700 m

Fig. 14. Estimated route information (black line) and route information placed by overlaying the aerial photo on the traversable area map (gray line).

Table 2. Error of estimated route position.

Distance from origin [m]	Distance error [m]		Angle error [rad]
200	(A1, B1) 1.35	(A2, B2) 1.75	−0.038
800	(A3, B3) 0.84	(A4, B4) 0.78	−0.00088
1200	(A5, B5) 1.97	(A6, B6) 2.03	0.0081
1700	(A7, B7) 1.42	(A8, B8) 1.06	−0.018

gation.

In the following section, we discuss whether each evaluation item is a factor in obstructing autonomous navigation at these points.

5.3. Discussion

The major factors that caused discontinuation of autonomous navigation can be divided into two categories: 1) insufficient recognition capacity of traversable areas and 2) the generation of discontinuity errors in SLAM-based localization.

The points where a low environment recognition capacity, i.e., item (3), is the cause are the 180 m and 210 m points (**Figs. 12(b)** and **(c)**, respectively). YVT-35LX detects the presence of low steps based on the difference between the highest and lowest points in the grid. The steps at these points are of height 5 cm; as this is lower than the value used for step recognition, the steps are undetected. Incidentally, the localization errors at these points are in the vicinity of 1.35 m and 1.75 m, respectively (**Table 2**); these are the distances (A1, B1) and (A2, B2) measured in the vicinity of the 200 m point shown in **Fig. 14(a)**. These errors are likely to have caused the robot to stray from the traversable area; however, the route estimation error is likely to be lower if the obstacle recognition is effectively carried out.

At the 130 m, 1,080 m, and 1,180 m points (shown in **Figs. 12(a)**, **(d)**, and **(e)**, respectively), a discontinuity of the self-position occurs; it results in its coincidence with an obstacle area as a result of SLAM localization. This prevents DWA-based route planning and results in dis-

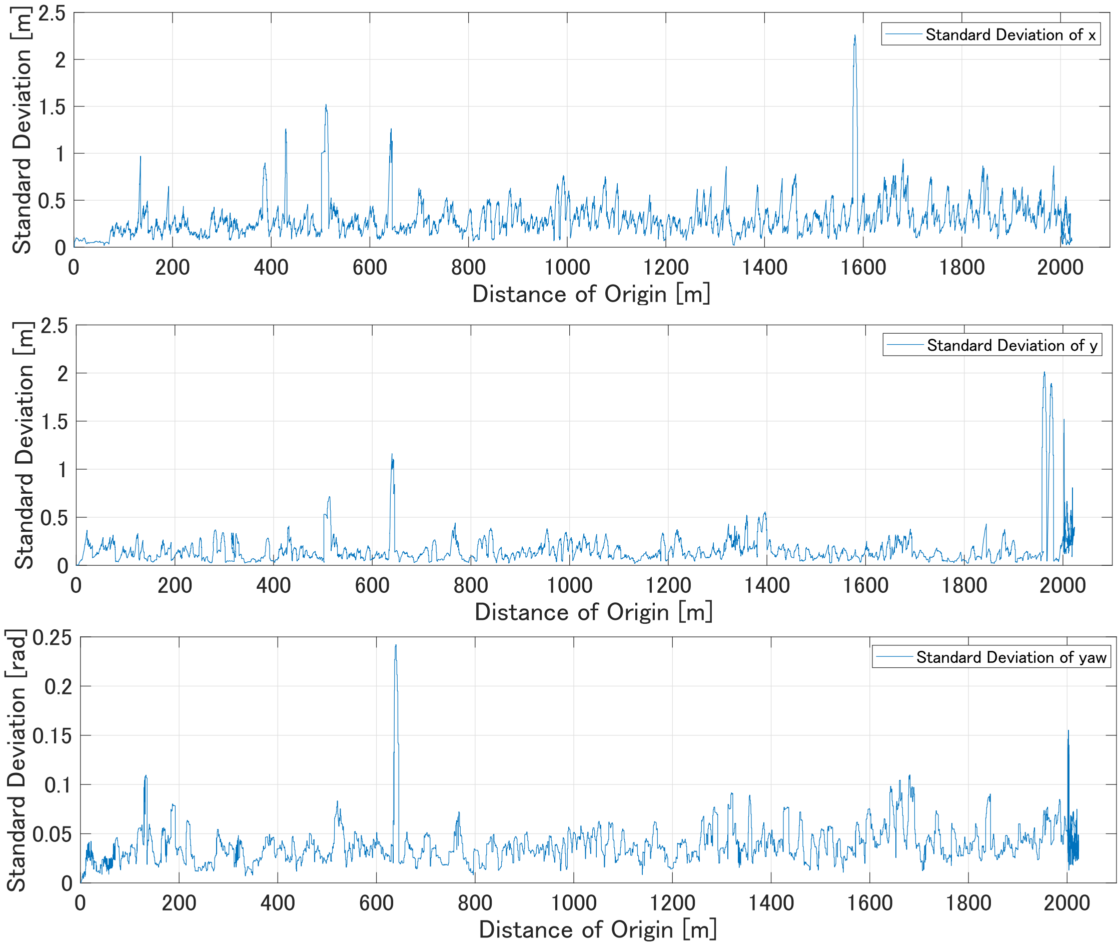
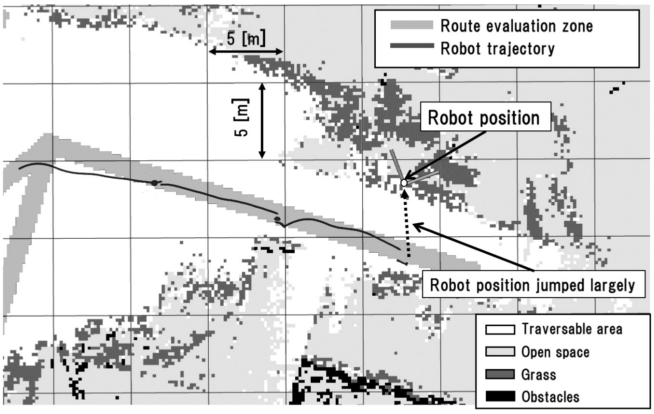


Fig. 15. Standard deviation of multiple odometry based on SLAM.

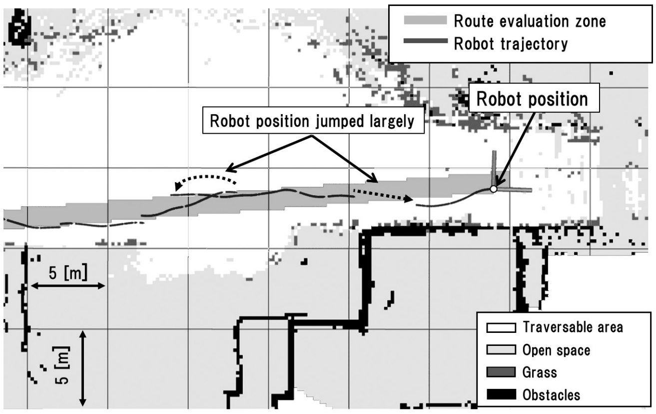
continuation. The self-position is estimated to lie on the obstacle area because the landmarks, which form repetitious patterns on the SLAM map, are scattered about at these locations and thus, are mismatched; this produces errors in the SLAM localization. This localization error causes the robot to locate its self-position on the obstacle area, rendering it unfeasible to carry out route planning with DWA. Because the SLAM accuracy is not particularly low at these points (**Fig. 15**), we can conclude that marginal discontinuities can disrupt a run. In these cases, the robot is able to move away from the obstacle area and continue autonomous navigation when it is manually operated to go forward a marginal distance. This indicates that the robot gradually recovers the correct self-position.

The 1,580 m, 1,600 m, 1,615 m, and 1,690 m points, shown in **Figs. 12(f)**, **(g)**, **(h)**, and **(i)**, respectively, lie in the section between 1,600 m and 1,700 m from the origin. **Fig. 15** indicates that these points are where the SLAM accuracy reduced. A considerable change in the SLAM self-position affects the geometric accuracy of the generated map. In particular, the position error that occurs at the 1,580 m point is likely to have resulted in a discontinuous map deformation. **Fig. 16** shows the plotted localized position on the SLAM map. Although the self-position on the generated map should be accurate and the trajectory should be continuous when SLAM functions correctly, discontinuous jumps are apparent at certain points. In particular, a jump of approximately 5 m to the side from the direction of travel occurs in **Fig. 16(a)**; this is considered to have caused the discontinuation of the run. Here, the SLAM-generated map undergoes discontinuous deformation because an erroneous matching of SLAM localization results in map deformation by referencing a point where landmarks are scarce. This phenomenon occurs because the accumulated landmark positions can be subsequently modified in slam_gmapping. The deformation of the map owing to mismatching results in loss of geometric accuracy; this caused the map to be overlaid inaccurately on the geometrically accurate route and therefore, erroneous route position estimation. Because the route evaluation zone is 150 m in length, the robot gradually ceases to use distorted sections as reference points beyond the 1,700 m point (which is 150 m further from the point where the problem occurs) and resumes normal navigation without the operator's manual intervention. This indicates that the length of the route evaluation zone, noted above as evaluation item (4), affects the recovery time in cases when the map becomes distorted owing to SLAM positioning error.

The above factors that caused navigation to be aborted are summarized in **Table 3**.



(a) 1600 m



(b) 1950 m

Fig. 16. Inconsistently estimated position during SLAM: lateral error of 5 m caused stack as shown in **Fig. 12(f)**. Longitudinal error in (b) did not cause stack.

Table 3. Causes of navigation failure.

Location (indicated in **Fig. 12**)	Cause of navigation failure
(b), (c)	Failure to detect obstacle (item (3), partly item (1)).
(a), (d), (e)	Inconsistent localization by SLAM (item (2)).
(f), (g), (h), (i)	Inaccurate localization by SLAM and map deformation (item (2), partly item (4)).

6. Conclusions

To develop a navigation system that enables a robot to autonomously navigate and arrive at its destination in locations it has not encountered before, this study proposes a method of matching metric route information and the map generated by SLAM based on the detection of traversable areas. Without surveying the actual site of the route, the robot operator is capable of making the robot navigate autonomously as long as sources to produce a metrically accurate map (e.g., aerial photos and architectural plans) are available.

Although there are sections in which the robot is incapable of continuing navigation, we would experimentally verify that the robot is capable of navigating most of the 2 km route set up in an actual outdoor environment, including a pedestrian walk. The discussions yielded the following conclusions:

- The consistency of SLAM, such as (the absence of) jumps in SLAM localization, substantially affects the performance of the method.
- In route estimation, a valid measure is to incorporate adjustments by including characteristic route topologies such as corners.
- Because the proposed method allows for positioning errors equal to approximately the road width, the robot's capacity to correctly recognize traversable areas has a substantial impact.

As an issue for future study, we consider it important to address cases in which sufficient metric route information cannot be obtained in advance. Whereas the aerial photos used in this study contain sections where the route cannot be visually verified owing to obstructions by buildings or trees, the route is selected based on reasonable estimates. We consider addressing such cases by excluding those visually obstructed sections from the route evaluation zone for route position estimation and carrying out route estimation by using only those visually verified sections so as to preserve geometric accuracy.

References:

[1] A. Suzuki, H. Date, T. Tsubouchi, A. Ohya, K. Shigematsu, and R. Mitsudome, "Development of Navigation System Based on Metric Route Information," Proc. of the 16th SICE SI Division Annual Conf., pp. 851-854, 2015 (in Japanese).

[2] A. Suzuki, H. Date, T. Tsubouchi, A. Ohya, K. Shigematsu, and R. Mitsudome, "Mobile Robot Navigation Using Metric Route Information and SLAM," Proc. of the 21st Robotics Symposia, pp. 256-261, 2016 (in Japanese).

[3] A. Suzuki, H. Date, T. Tsubouchi, A. Ohya, K. Shigematsu, and R. Mitsudome, "Navigation of a Mobile Robot Based on Metric Route Map at Places to Visit for the First Time," Proc. of 2016 JSME Conf. on Robotics and Mechatronics, 2A2-07a7, 2016 (in Japanese).

[4] A. Suzuki, H. Date, T. Tsubouchi, A. Ohya, K. Shigematsu, and R.Mitsudome, "Navigation Dependent Solely on Metric Route Information as Prior Information," Proc. of the 17th SICE SI Division Annual Conf., pp. 117-121, 2016 (in Japanese).

[5] S. Thrun, W. Burgard, and D. Fox, "Probabilistic Robotics," MIT Press, 2007.

[6] P. Ranganathan, J. B. Hayet, M. Devy, S. Hutchinson, and F. Lerasle, "Topological navigation and qualitative localization for indoor environment using multi-sensory perception," Robotics and Autonomous Systems, Vol.41, pp. 137-144, 2002.

[7] S. Muramatsu, T. Tomizawa, S. Kudoh, and T. Suehiro, "Mobile Robot Navigation Utilizing the WEB Based Aerial Images Without Prior Teaching Run," J. Robot. Mechatron., Vol.29, No.4, pp. 697-705, 2017.

[8] K. Irie, M. Sugiyama, and M. Tomono, "Localization Using 2D Street Maps in Unfamiliar Environments," Proc. of the 20th Robotics Symposia, pp. 157-164, 2015 (in Japanese).

[9] P. Ruchti, B. Steder, M. Ruhnke, and W. Burgard, "Localization on OpenStreetMap data using a 3D laser scanner," Proc. the IEEE Int. Conf. on Robotics and Automation, pp. 5260-5265, 2015.

[10] T. Ort, L. Paull, and D. Rus, "Autonomous vehicle navigation in rural environments without detailed prior maps," Int. Conf. on Robotics and Automation, 2018.

[11] A. Watanabe, S. Bando, and S. Yuta, "Road Following Based Outdoor Navigation by Detecting Orientation and Finding Intersection," Proc. of the 12th SICE SI Division Annual Conf., pp. 1773-1776, 2011 (in Japanese).

[12] A. Watanabe, "Research of a High Performance Measurement and Control System for Mobile Robot Based on Offline Calculation Support," Heisei 26th academic year, Ph.D. Dissertations in Graduate School of Systems and Information Engineering, Department of Computer Science, University of Tsukuba, 2014 (in Japanese).

[13] Y. Hosoda, R. Sawahashi, N. Machinaka, R. Yamazaki, Y. Sadakuni, K. Onda, R. Kusakari, M. Kimba, T. Oishi, and Y. Kuroda, "Robust Road-Following Navigation System with a Simple Map," J. Robot. Mechatron., Vol.30, No.4, pp. 552-562, 2018.

[14] H. Ikeda, Y. Kobayashi, and K. Hirano, "How car navigation systems have been put into practical use," Synthesiology, Vol.3, No.4, pp. 292-300, 2010 (in Japanese).

[15] K. Demura and Y. Komoriya, "A Navigation Method using the Mutual Feedback of Way-points and Self-positions," Advanced Robotics, Vol.26, pp. 1677-1691, 2012.

[16] G. Grisetti, C. Stachniss, and W. Burgard, "Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters," IEEE Trans. on Robotics, Vol.23, pp. 34-46, 2007.

[17] K. M. Wurm, R. Kmmerle, C. Stachniss, and W. Burgard, "Improving Robot Navigation in Structured Outdoor Environments by Identifying Vegetation from Laser Data," Proc. The 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 1217-1222, 2009.

[18] H. Date, S. Ohkawa, Y. Takita, and J. Kikuchi, "High Precision Localization of Mobile Robot Using LIDAR Intensity of Surface," Trans. of the Japan Society of Mechanical Engineers, C, Vol.79, No.806, pp. 100-109, 2013 (in Japanese).

[19] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," IEEE Trans. on Robotics and Automation, Vol.4, No.1, pp. 23-33, 1997.

[20] H. Date and Y. Takita, "Real World Experiments of an Autonomous Mobile Robot in the Pedestrian Environment," Proc. the 5th Int. Conf. on Automation, Robotics and Applications, pp. 413-418, 2011.

Supporting Online Materials:

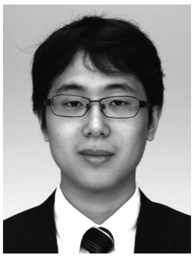
[a] Tsukuba Challenge
http://www.tsukubachallenge.jp/ [Accessed October 22, 2018]

Name:
Asahi Handa

Affiliation:
Master Student, Department of Intelligent Interaction Technologies, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
2018- Master Student, University of Tsukuba



Name:
Azumi Suzuki

Affiliation:
Lead Engineer, Tier IV Inc.

Address:
1-1-3 Mei-eki, Nakamura-ku, Nagoya, Aichi 450-6627, Japan

Brief Biographical History:
2017- Honda R&D Co., Ltd.
2018- Tier IV Inc.

Name:
Hisashi Date

Affiliation:
Associate Professor, Faculty of Engineering, Information and Systems, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
2003-2013 Research Associate, National Defense Academy of Japan
2005-2006 Visiting Associate, California Institute of Technology
2013-2015 Lecturer, National Defense Academy of Japan
2015- Associate Professor, University of Tsukuba

Main Works:
• "Autonomous Mobile Robot Searching for Persons with Specific Clothing on Urban Walkway," J. Robot. Mechatron., Vol.29, No.4, pp. 649-659, Aug. 2017.

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

Name:
Ryohsuke Mitsudome

Affiliation:
Software Engineer, Tier IV Inc.

Address:
1-1-3 Mei-eki, Nakamura-ku, Nagoya, Aichi 450-6627, Japan

Brief Biographical History:
2018- Honda Moter Co., Ltd.
2018- Tier IV Inc.

Main Works:
• "Autonomous Mobile Robot Searching for Persons with Specific Clothing on Urban Walkway," J. Robot. Mechatron., Vol.29, No.4, pp. 649-659, 2017.

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

Name:
Takashi Tsubouchi

Affiliation:
Professor, Faculty of System and Information Engineering, University of Tsukuba

Address:
1-1-1 Tennoudai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
1989- Assistant Professor, Utsunomiya University
1991- Assistant Professor, The University of Tokyo
1994- Lecturer, University of Tsukuba
1998- Associate Professor, University of Tsukuba
2006- Professor, University of Tsukuba

Main Works:
• "Forest Mapping and Trunk Parameter Measurement on Slope Using a 3D-LIDAR," 2017 IEEE/SICE Int. Symp. on System Integration, Taipei, 2017.
• "Rock moving operation using a rod-shaped manipulator," J. Robot. Mechatron., Vol.25, No.1, pp. 232-239, 2013.

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

Name:
Akihisa Ohya

Affiliation:
Professor, Department of Information Engineering, Faculty of Engineering, Information and Systems, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
1992-1994 Research Associate, University of Tsukuba
1994-2000 Assistant Professor, University of Tsukuba
1995-1997 Visiting Scholar, Purdue University
2000-2012 Associate Professor, University of Tsukuba
2000-2003 Researcher, PRESTO, JST
2012- Professor, University of Tsukuba

Main Works:
• "Development of Small Size 3D LIDAR," 2014 IEEE Int. Conf. on Robotics and Automation, pp. 4620-4626, May 2014.
• "Image Correspondence Based on Interest Point Correlation in Difference Streams: Method and Applications to Mobile Robot Localization," J. Robot. Mechatron., Vol.28, No.2, pp. 234-241, Apr. 2016.

Membership in Academic Societies:
• The Institute of Electrical and Electronic Engineers (IEEE)
• The Robotics Society of Japan (RSJ)
• The Society of Instrument and Control Engineers (SICE)
• The Japan Society of Mechanical Engineering (JSME)
• The Institute of Electronics, Information and Communication Engineers (IEICE)
• The Acoustical Society of Japan (ASJ)