Stereo-Based Outdoor Localization using a Line Drawing Building Map

Keisuke Matsuo Jun Miura Junji Satake
Department of Computer Science and Engineering
Toyohashi University of Technology, Toyohashi, Japan

***Abstract*— Line drawing maps are frequently used by people to exchange location information. Mobile robots in the future are expected to be able to communicate with people and thus to localize themselves using such maps. We therefore develop a method of outdoor localization using a stereo camera and a line drawing building map. Omnidirectional range data obtained by a stereo camera and a pan-tilt unit are projected onto a robot-centered 2D grid map, which is to be matched with an input line drawing map. Since various objects such as trees often obstruct buildings in usual outdoor scenes, we apply a view-based object classifier in order to extract only stereo data from buildings. To cope with uncertainties in stereo data and building extraction, we adopt Monte Carlo localization. Experimental results show that our simple approach works reasonably well in an actual outdoor environment.**

***Index Terms*— Outdoor navigation, Line-drawing map, Stereo, Particle filter.**

I. Introduction

Localization is one of the fundamental functions of mobile robots. People often use a line drawing map and can navigate themselves where they have never been. Although only a part of objects such as buildings are written in the map, people can determine their position with respect to such objects by matching their knowledge of the map with observations of the environment. Since mobile robots in the future are expected to communicate with people as in the way people do, localization using a line drawing map will be an important capability. As a first step towards this goal, this paper proposes an outdoor localization method using line drawing building maps and stereo.

In recent times, vision is often used for SLAM and localization problems [13, 5, 4, 6], in which visual features are used as landmarks. Several view-based approaches have also been proposed [1, 11]. These works are basically learning-based, that is, the localization is performed using a learned map. This means that the data of the environment should be collected *before* localization.

Yun and Miura [15] proposed a localization method using a line drawing building map with uncertainty. Their method relies on line segment features constituting building boundaries, the directions of which are estimated using vanishing points, and stereo range features corresponding to building walls. Since many false matches exist between the features and a map, they adopt a multi-hypothesis approach, which is relatively complicated and costly. Leung, Clark, and Huissoon [8] proposed a localization method based on the matching between such line segments and the line features extracted from aerial images in conjunction with particle filter. Kümmerle et al. [7] developed a graph SLAM method based on a similar idea and an accurate 3D range sensor. Parsley and Julier [12] proposed a general framework of exploiting the use of various prior information with uncertainty in SLAM.

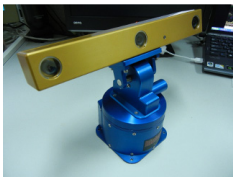
In usual outdoor scenes, buildings are sometimes good landmarks but often obstructed by various objects such as trees and bushes. Conventional SLAM method using 2D range finders (e.g., [3, 2]) are therefore most probably difficult to apply. The above methods using prior information employ several heuristics to extract information from the input data (from a 3D laser scanner, for example) and maps (or aerial images) to be matched.

In this paper, we develop a localization method using a stereo camera and a line drawing building map. We extract stereo range data from building by a view-based object classifier. The extracted data are matched with a part of the map, which is actually a set of visible building walls, in the Monte Carlo localization framework. We use omnidirectional stereo data acquired by a stereo camera on a pan-tilt head to obtain a wide field of view. We will show our simple method works reasonably well in actual outdoor localization problems in spite of the low quality of stereo range data.

The rest of the paper is organized as follows. Sec. II describes our robot and acquisition of stereo data. Sec. III describes a mapping of stereo data to a local map with view-based building data extraction. Sec. IV explains the input line drawing map and hidden lines removal for matching. Sec. V describes the procedure for Monte Carlo localization. Sec. VI shows experimental results validating our approach. Sec. VII discusses the current performance and possible extensions of the proposed method, and Sec. VIII summarizes the paper.

II. Stereo data acquisition

A wider field of view is usually more effective for localization, especially in outdoor environments where the features are more scarce and distance to objects are larger. So we configure a measurement unit which has a stereo camera (Point Gray Research Bumblebee XB3, $66\,[deg.]$ horizontal field of view) mounted on a pan-tilt head (TRAC Labs Biclops PT) and acquire omnidirectional stereo data. We take pairs of stereo images at six panning positions while the robot is stopping. The tilt angle is set to $20\,[deg.]$ looking up. The measurement unit is put on a mobile robot which is based on an electric wheelchair (PatraFour from Kanto Auto Works



(a) measurement unit.

(b) Our mobile robot.

Fig. 1. The measurement unit and the robot.

Fig. 2. An example set of six input images.

Co.). Fig. 1 shows the measurement unit and the mobile robot. Fig. 2 shows an example set of six images, collectively cover $360\,[deg.]$, taken at a position.

Fig. 3 is a mapping of omnidirectional stereo data onto a floor in a relatively small room; stereo measurements are fairly accurate in small distances. For distant objects, however, range measurements will have large errors as shown later.

III. Local Mapping of Building Data

A set of stereo data obtained at a robot position is transformed into a local grid map. An SVM (support vector machine)-based building classifier is used for extracting only stereo data from buildings.

A. Extracting building data using SVM-based classifier

We use the SVM-based classifier proposed by Miura and Yamamoto [11], which has been shown to be effective in view-based localization under various weather and seasons.

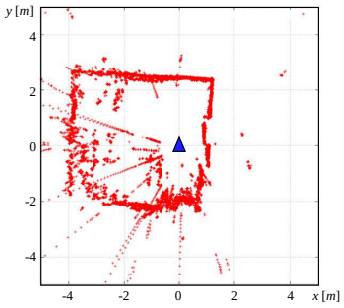


Fig. 3. An example indoor mapping using the measurement unit.

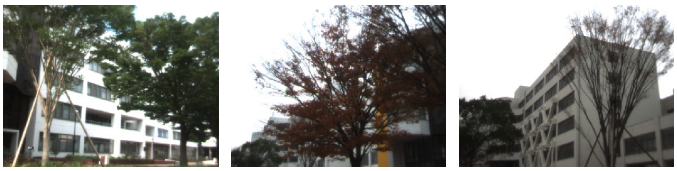


Fig. 4. Examples of training images.

Fig. 5. Building extraction results for the images shown in Fig. 2.

We divide an input image to 16x16 windows and classify each region using several image features. For the building classifier, we use the normalized color (r, g, b), an edge density, the peak value of the voting in Hough transform, and the variance of edge directions [11]. This classifier is for extracting building boundaries and windows where strong edge segments exist.

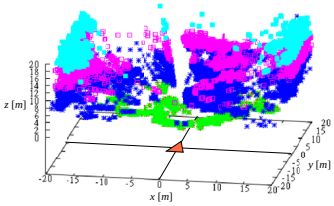
We used 57 training images taken in various seasons and weather conditions. The windows in the images are manually labeled for training. Some of the training images are shown in Fig. 4. We use the SVM with RBF kernel ($K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp(-\gamma||\boldsymbol{x}_1 - \boldsymbol{x}_2||^2)$, $\gamma = 50$) for this classifier.

Fig. 5 shows the building extraction results for the images shown in Fig. 2. Although some regions other than buildings are extracted, the overall result is reasonable because complete classification is not a necessary condition for Monte Carlo localization.

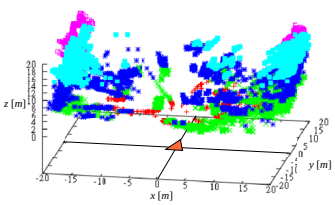
B. Making a local map

The map given to the robot is a 2D line drawing map. The obtained stereo data are therefore converted into a 2D local grid map for matching. The local map is robot-centered and the size of each cell is $0.1\,[m] \times 0.1\,[m]$. Each of stereo data points, which is originally represented in the camera coordinates, is transformed into the robot local coordinates and then voted on the grid map. Each cell accumulates the votes. Finally, a Gaussian smoothing is applied to the local map to consider the discretization of the grid map.

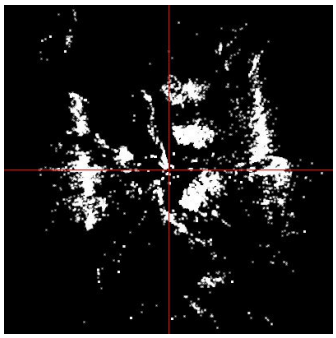
Stereo measurements have larger uncertainty for distant objects, and many models for this uncertainty have been proposed (e.g., [10]). In this mapping, however, we do not explicitly consider the positional error of each measured point. The reason is as follows. We model buildings by their 2D boundaries and each boundary is matched with the map in localization as described later. The distribution of *actual*



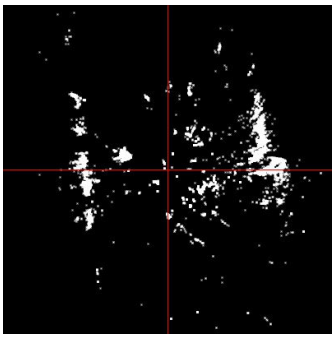
(a) Local map without building extraction.



(b) Local map with building extraction.



(c) Local map without building extraction.



(d) Local map with building extraction.

Fig. 6. Effect of building extraction on the local map.

measured points for a boundary is naturally larger for a more distant building and their contribution to localization will be smaller without any adjustments.

C. Mapping results

Fig. 6(a) is the 3D data points obtained from the images shown in Fig. 2. These data are projected and smoothed on a 2D plane to produce the local grid map shown in Fig. 6(c). The data from large trees in front of the building on the right (see the second and the third image in Fig. 2) exist in the map.

Fig. 6(b) and (d), on the other hand, indicate the 3D data points and the local grid map, respectively, after applying the building data extraction. These figures show that many of the data from large trees mentioned above are deleted correctly, although the 2D position of data points are not very accurate.

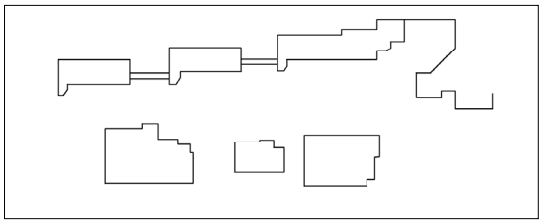
IV. Line Drawing Map and Hidden Lines Removal

A. Map representation

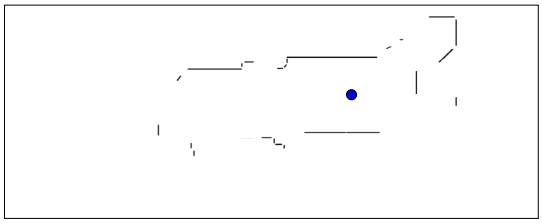
The input line drawing map is composed of lines, each of which is represented by a pair of 2D endpoints in the world coordinates. We developed a line drawing map editor for making, modifying, and saving the map data. The shapes of the buildings are manually measured from the floor and campus plans. The scale of the plans is 1:2500 and the error in the map is estimated to reach a few meters.

B. Hidden lines removal

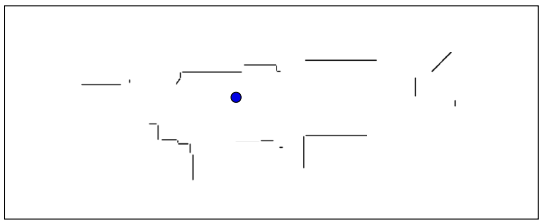
The degree of matching between the input local map and the line drawing map is used for calculating the likelihood of a particle in Monte Carlo localization, as described later. Although the input map includes all building boundaries, only a part of them is visible from a given position. It is therefore necessary to remove occluded boundaries from the position in order to increase the reliability of matching. Since the input map has only 2D information, we consider only 2D visibility and thus perform a hidden lines removal.



(a) Line drawing map.



(b) Hidden line removal example 1.



(c) Hidden line removal example 2.

Fig. 7. Examples of hidden lines removal. Circular markers in (b) and (c) indicate robot positions.

Once a robot position is set on the world coordinates, the visible part of each line segment can be calculated by analyzing the relationship between the robot position (i.e., viewpoint), that line segment, and other potentially occluding segments. Note that since we use omnidirectional stereo data, the orientation of the robot does not matter in this hidden lines removal. Fig. 7 shows examples of hidden lines removal for two different robot positions.

V. Localization

A. Monte Carlo localization

Monte Carlo localization (MCL) [14] is a powerful tool for localizing the robot under uncertain prior knowledge and sensor data, and has been successfully applied to various localization problems.

The state vector is represented by a 2D robot pose in the world coordinates, that is:

$$\boldsymbol{x} = (x_w, y_w, \theta_w)\,. \tag{1}$$

The algorithm of MCL is as follows:

1) **Initialization**. Particles are distributed around the starting position.
2) Repeat the following for each pair of movement and measurement:

a) **Prediction step**. Each particle is moved by the odometry reading with a probabilistic error sampling according to the odometry error estimate. The odometry error is estimated according to the motion model described below (see Sec. V-B).
b) **Correction step**. The likelihood of each particle is calculated as the degree of matching. We compare the local grid map with a part of line drawing map composed of only visible lines to estimate the likelihood (see Sec. V-C).
c) **Resampling step**. Resampling is performed by draw with replacements using the normalized weights as probabilities for particles.

B. Motion model

We calculate a new pose $(x_w^t, y_w^t, \theta_w^t)$ from the previous pose $(x_w^{t-1}, y_w^{t-1}, \theta_w^{t-1})$ and odometry data $(\Delta x_t, \Delta y_t, \Delta\theta_t)$ as follows:

$$\begin{pmatrix} x_w^t \\ y_w^t \\ \theta_w^t \end{pmatrix} = \begin{pmatrix} x_w^{t-1} \\ y_w^{t-1} \\ \theta_w^{t-1} \end{pmatrix} + \begin{pmatrix} \cos\theta_w^{t-1} & -\sin\theta_w^{t-1} & 0 \\ \sin\theta_w^{t-1} & \cos\theta_w^{t-1} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \Delta x_t \\ \Delta y_t \\ \Delta\theta_t \end{pmatrix} + \begin{pmatrix} n_x \\ n_y \\ n_\theta \end{pmatrix}, \quad (2)$$

where (n_x, n_y, n_θ) is the process noise. Currently, each element of the noise estimate is independent of each other and constant; the standard deviation of n_x, n_y, and n_θ for the movement of $1\,[m]$ is set to $0.05\,[m]$, $0.05\,[m]$, and $1.0\,[deg.]$, respectively.

C. Likelihood calculation

The likelihood of each particle is given by the degree of matching between the local grid map and the input line drawing map with hidden lines removal. Using the position of a particle, lines visible from that point are calculated and mapped onto the local map using the pose of the particle. The values in the local map are first summed up where the mapped lines exist and then divided by the number of pixels on the lines to calculate the averaged value, which is used as the degree of matching.

Another approach to calculating the likelihood might be extracting line features from the local map and to match them with the line drawing map. We, however, do not take this approach because a reliable segmentation of measured points into boundaries and an accurate line fitting seem difficult due to large stereo measurement errors.

VI. Experimental results

A. Experimental environment

Fig. 8 shows the experimental site in our campus and two routes used. The buildings on both sides of the routes are from two to eight stories high, although that information is not written in the line drawing map. We moved the robot along the routes using a Wiimote and took data for off-line localization. The rough estimate of the initial position is given to the robot, but the orientation is not.

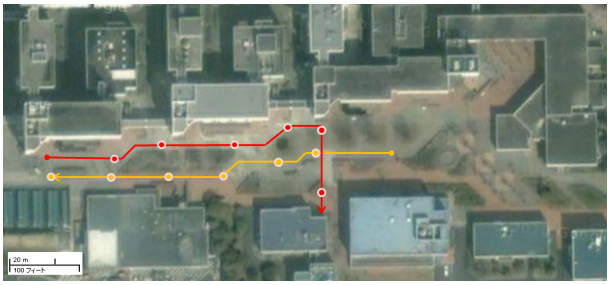


Fig. 8. Route for the experiments.

B. Localization results

Figs. 9 and 10 show the results at several positions on the red and the orange route in Fig. 8, respectively. Several factors, in general, affect the degree of convergence of particles. One is the distance to the surrounding buildings because the uncertainty in stereo range data become larger for more distant objects. Another factor is the existence of objects obstructing building views. Although our building extraction works reasonably well, it is not complete and uneliminated stereo data degrade the localization quality.

In Fig. 9, particles are well converged because the robot moves near the building on the left (see the red route in Fig. 8). On the other hand, in Fig. 10, the particles diverge more at many places because the route is in the middle of the buildings and is near big trees (see the orange route in Fig. 8). Although the degree of convergence changes from place to place, the overall localization ability is reasonable considering the uncertainty in stereo data and building extraction.

We then performed localization using raw stereo data without building extraction (see Fig. 6(c)). Fig. 11 shows the results. Compared to those in Fig. 10, the localization accuracy is much worse in places with trees (steps 5, 10, and 20). These results show the effectiveness of building extraction in tree-populated areas.

VII. Discussion

A. Localization accuracy

Necessary accuracy in localization depends on the objective of each mobile robot and its task. If the robot has to reach a specific goal only by the navigation method like the one presented in this paper, the accuracy should be, say, around $50\,[cm]$. This is a hard task with only stereo range data. Such a high accuracy is not, however, necessarily required when we view the navigation task hierarchically.

Let us consider the following scenario. The task of the robot is to reach the entrance of some building. Then, a global-level navigation takes the robot to the place where it can observe the entrance, and a local-level navigation guides the robot movement to there using visual feedback. Similar scenarios can be applied to navigation in urban areas, where signboards and logos of shops are used for final, precise localization.

In such a scenario, the navigation presented in this paper corresponds to the global-level one, and the required accuracy is not very high; a required accuracy is to assure that the robot

can observe the entrance after the global-level navigation. Our proposed approach will suffice for this purpose.

Also note that even if a global-level navigation provides very accurate positioning, it is still necessary to be combined with some local scene recognition methods such as road and obstacle detection. We are now planning to combine the localization method presented in this paper with our multisensory road detection method [9].

B. Map uncertainty

At present, line drawing maps are assumed to be almost complete except that objects other than buildings are not written. The current localization method relies on this completeness. On the other hand, people can navigate themselves using hand-drawn maps which include many errors, omissions, and sometimes large deformations. In [16], we defined four *key uncertainties* usually included in hand-drawn maps: dimension, position, shape, and existence uncertainty. As far as the amount of uncertainty is small enough, the current method will probably work thanks to the MCL approach, but will not for larger uncertainties, especially for existence uncertainties (i.e., omission of many buildings in the map).

We sometimes use maps with drastic omissions of buildings, but instead, we usually put additional landmark information such as prominent buildings. It is therefore necessary to improve the method to utilize such additional information in localization.

C. Data acquisition cost and localization accuracy

Taking omnidirectional stereo data is certainly the most costly part of the current system (about six seconds, at present), although it provides fairly accurate stereo data thanks to a well-calibrated stereo camera. Motion stereo for omnidirectional images can be used for obtaining point clouds, while it may degrade the position data. It is also possible to adopt various SLAM or visual odometry algorithms to obtain reliable position data, but it is still important to extract only information written in the map, as done by a view-based classifier in this paper.

VIII. Summary

This paper describes an outdoor localization method using a stereo camera and a line drawing building map. We extract stereo data from building using an SVM-based classifier to be matched with visible building boundaries on the map. The Monte Carlo localization approach provides a reasonably good localization performance in spite of incomplete building extraction and low quality stereo range data. Several future research directions have also been discussed.

Acknowledgment

This work is supported in part by Grant-in-Aid for Scientific Research (No. 21300075) from JSPS.

References

[1] M. Cummins and P. Newman. FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. *Int. J. of Robotics Research*, Vol. 27, No. 6, pp. 647–665, 2008.

[2] G. Grisetti, C. Stachniss, and W. Burgard. Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling. In *Proceedings of 2005 IEEE Int. Conf. on Robotics and Automation*, pp. 2432–2437, 2005.

[3] J.E. Guivant and E.M. Nebot. Optimization of the Simultaneous Localization and Map-Building Algorithm for Real-Time Implementation. *IEEE Trans. on Robotics and Automation*, Vol. 17, No. 3, pp. 242–257, 2001.

[4] K. Irie, T. Yoshida, and M. Tomono. Mobile Robot Localization Using Stereo Vision in Outdoor Environments under Various Illumination Conditions. In *Proceedings of 2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 5175–5181, 2010.

[5] J. Klippenstein and H. Zhang. Quantitative Evaluation of Feature Extractors for Visual SLAM. In *4th Canadian Conf. on Computer and Robot Vision (CRV '07)*, pp. 157–164, 2007.

[6] M. Kronfeld, C. Weiss, and A. Zell. Swarm-supported Outdoor Localization with Sparse Visual Data. *Robotics and Autonomous Systems*, Vol. 50, No. 2, pp. 166–173, 2010.

[7] R. Kümmerle, B. Steder, C. Dornhege, A. Kleiner, G. Grisetti, and W. Burgard. Large Scale Graph-Based SLAM using Aerial Images as Prior Information. *Autonomous Robots*, Vol. 30, No. 1, pp. 25–39, 2011.

[8] K.Y.K. Leung, C.M. Clark, and J.P. Huissoon. Localization in Urban Environments by Matching Ground Level Video Images with an Aerial Image. In *Proceedings of 2008 IEEE Int. Conf. on Robotics and Automation*, pp. 551–556, 2008.

[9] Y. Matsushita and J. Miura. On-Line Road Boundary Modeling with Multiple Sensory Features, Flexible Road Model, and Particle Filter. *Robotics and Autonomous Systems*, Vol. 59, No. 5, pp. 274–284, 2011.

[10] J. Miura and Y. Shirai. Vision and Motion Planning for a Mobile Robot under Uncertainty. *Int. J. of Robotics Research*, Vol. 16, No. 6, pp. 806–825, 1997.

[11] J. Miura and K. Yamamoto. Robust View Matching-Based Markov Localization in Outdoor Environments. In *Proceedings of 2008 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2970–2976, 2008.

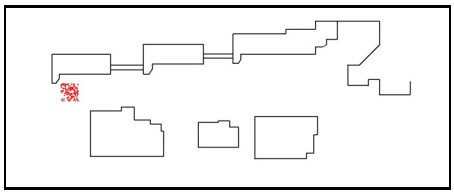
[12] M.P. Parsley and S.J. Julier. Towards the Exploitation of Prior Information in SLAM. In *Proceedings of 2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2991–2996, 2010.

[13] F.T. Ramos, J. Nieto, and H.F. Durrant-Whyte. Recognising and Modeling Landmarks to Close Loops in Outdoor SLAM. In *Proceedings of 2007 IEEE Int. Conf. on Robotics and Automation*, pp. 2036–2041, 2007.

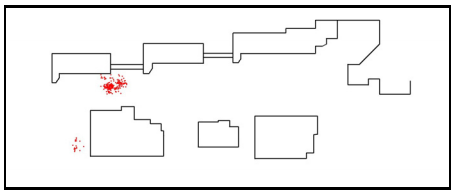
[14] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. The MIT Press, 2005.

[15] J. Yun and J. Miura. Multi-Hypothesis Localization with a Rough Map using Multiple Visual Features for Outdoor Navigation. *Advanced Robotics*, Vol. 21, No. 11, pp. 1281–1304, 2007.

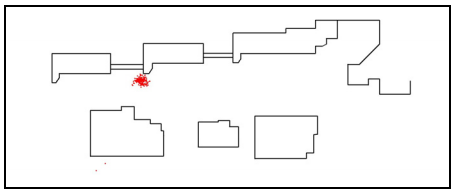
[16] J. Yun and J. Miura. A Quantitative Navigability Measure of Rough Maps. *J. of Robotics and Mechatronics*, Vol. 21, No. 1, pp. 95–103, 2009.



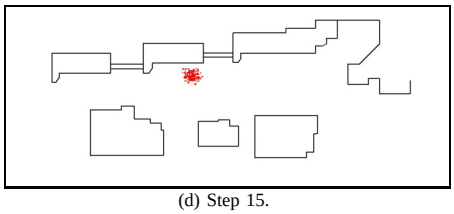
(a) Step 0.



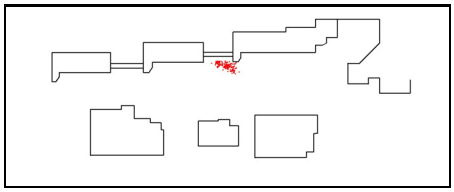
(b) Step 5.



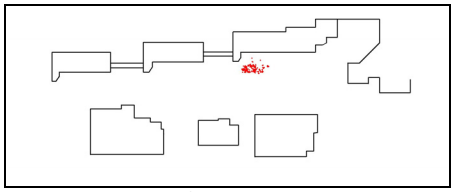
(c) Step 10.



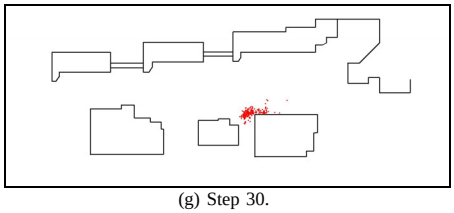
(d) Step 15.



(e) Step 20.

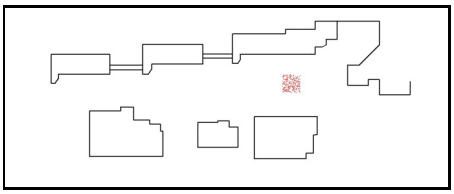


(f) Step 25.

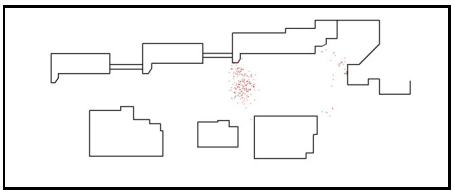


(g) Step 30.

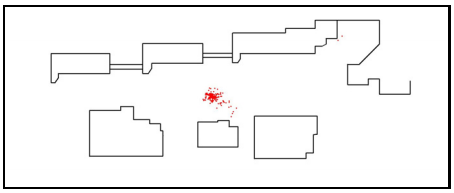
Fig. 9. Localization result for the *red* route in Fig. 8.



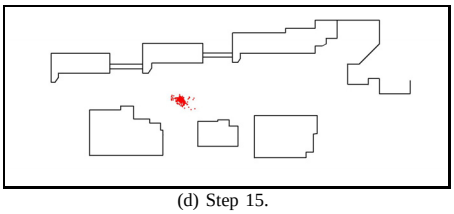
(a) Step 0.



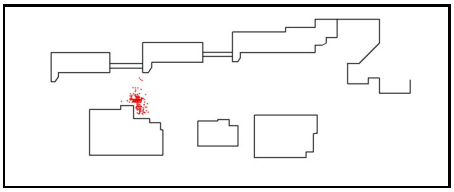
(b) Step 5.



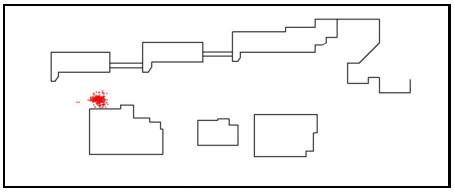
(c) Step 10.



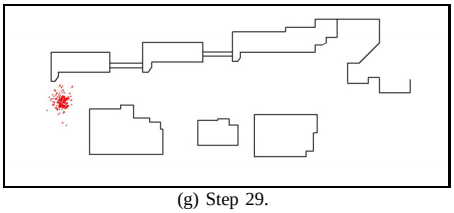
(d) Step 15.



(e) Step 20.

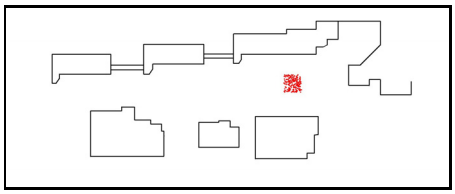


(f) Step 25.

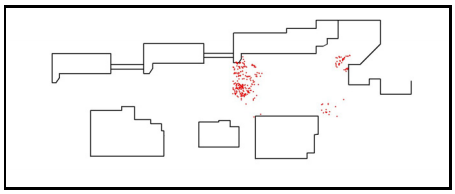


(g) Step 29.

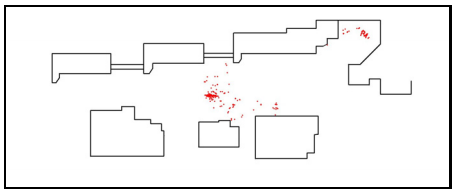
Fig. 10. Localization result for the *orange* route in Fig. 8.



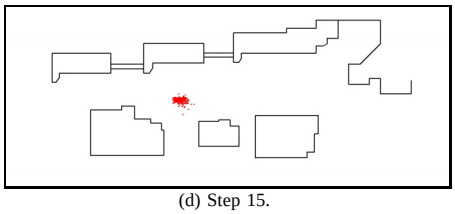
(a) Step 0.



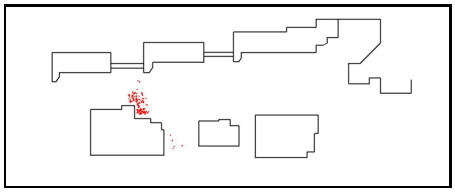
(b) Step 5.



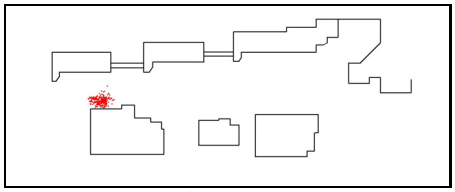
(c) Step 10.



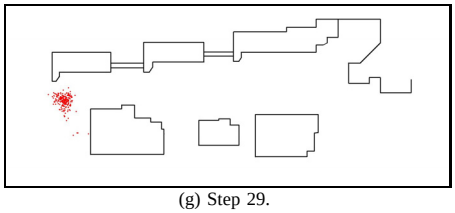
(d) Step 15.



(e) Step 20.



(f) Step 25.



(g) Step 29.

Fig. 11. Localization result for the *orange* route in Fig. 8 without building extraction.