SeqSLAM++: View-based Robot Localization and Navigation

Shuji Oishi[a,*], Yohei Inoue[b], Jun Miura[b], Shota Tanaka[b]

[a] *Smart Mobility Research Team, Robot Innovation Research Center, Department of Information Technology and Human Factors, National Institute of Advanced Industrial Science and Technology (AIST)*
[b] *Department of Computer Science and Engineering, Toyohashi University of Technology*

Abstract

This paper presents a new approach to view-based localization and navigation in outdoor environments, which are indispensable functions for mobile robots. Several approaches have been proposed for autonomous navigation. GPS-based systems are widely used especially in the case of automobiles, however, they can be unreliable or non-operational near tall buildings. Localization with a precise 3D digital map of the target environment also enables mobile robots equipped with range sensors to estimate accurate poses, but maintaining a large-scale outdoor map is often costly. We have therefore developed a novel view-based localization method *SeqSLAM++* by extending the conventional SeqSLAM in order not only to robustly estimate the robot position comparing image sequences but also to cope with changes in a robot's heading and speed as well as view changes using wide-angle images and a Markov localization scheme. According to the direction to move provided by the SeqSLAM++, the local-level path planner navigates the robot by setting subgoals repeatedly considering the structure of the surrounding environment using a 3D LiDAR. The entire navigation system has been implemented in the ROS framework, and the effectiveness and accuracy of the proposed method was evaluated through off-line/on-line navigation experiments.

Keywords: SeqSLAM, View-based localization, Navigation, Mobile robot

1. Introduction

The mobile service robot is an emerging application area in robotics. Such a robot is expected to provide various service tasks like attending, guiding, and searching. One of the necessary functions of mobile service robots is *navigation*, which makes it possible for a robot to move from one place to another autonomously. Since outdoor environments are an important part of human activity, mobile service robots should be able to navigate themselves in the outdoors.

Outdoor navigation can be divided into two levels. The global level deals with localization and subgoal selection, while the local level deals with safe navigation in a local area. This is an analogy to a navigated car driving; a car navigation system tells a driver where the car is and which way to go, and the driver is responsible for safe driving, including following traffic rules and avoiding possible collisions.

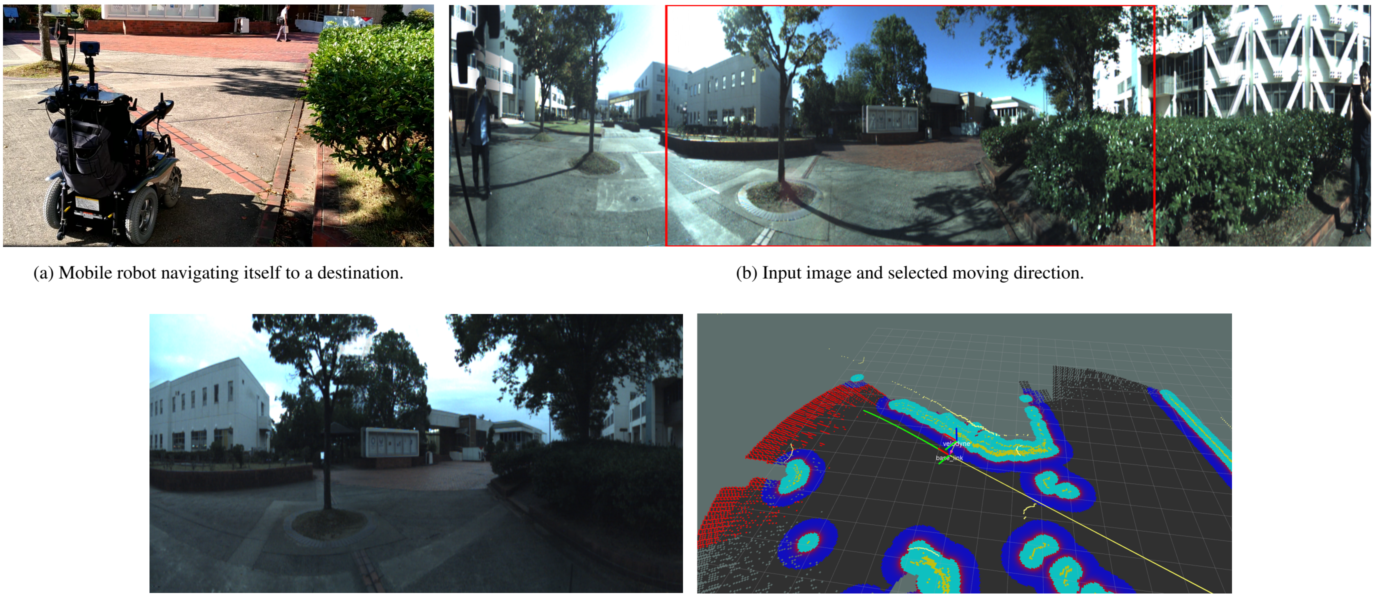
Several approaches are possible for outdoor global localization. GPS-based systems are often used [18, 19], but can be unreliable or non-operational near tall buildings. A precise digital map of the environment is also required in this approach. A map-based approach is also popular in the SLAM context [20, 21, 22], however making a large-scale outdoor map is often costly. In this paper, we seek a simpler way, that is, view-based localization [1, 2, 3, 4].

In a typical view-based localization, a route to follow is represented by a sequence of views and a view-matching procedure is employed for determining the robot's current location. One of the issues when applying this approach to outdoor scenes is robustness to view changes due to changes of, for example, weather, seasons, and time of day. Among various approaches to resolving the issue, we adopt the SeqSLAM method [9], which achieves a high robustness even under extreme view changes by using image sequence matching. Although this method shows a good performance for image sequences taken from a vehicle, it is not always directly applicable to mobile robot navigation.

We therefore developed *SeqSLAM++* by extending the conventional SeqSLAM for a new view-based navigation framework for mobile robots. The major contribution of this paper is to generalize the SeqSLAM so that it can handle frequent changes of a robot's heading and speed due to its motion, and provide a direction to move at a certain interval to follow the training trajectory. As shown in Fig. 1, the proposed SeqSLAM++ autonomously navigates the mobile robot to the destination according to the result of the robust image sequence matching. Combined with local mapping and path planning capabilities, the SeqSLAM++ generates a local subgoal based on the estimated direction to move and makes the mobile robot go to the destination tracing each subgoal.

The rest of the paper is organized as follows. Sec. 2 presents related work on view-based localization. Sec. 3 describes the detail of the proposed SeqSLAM++ for global-level mobile robot localization and navigation. Sec. 4 describes a local navigation strategy including local mapping, subgoal selection, and path planning. Sec. 5 shows the experimental results of off-line

*Corresponding author
Email address: `shuji.oishi@aist.go.jp` (Shuji Oishi)



(a) Mobile robot navigating itself to a destination.

(b) Input image and selected moving direction.

(c) Estimated location (The best matched image in the training sequence).

(d) Local path planning along with the direction to move.

Figure 1: Autonomous navigation with the proposed SeqSLAM++.

performance validation, on-line navigation, and trajectory evaluation. Sec. 6 concludes the paper and discusses future work.

2. Related Work

View-based localization is to find the location of a robot or a vehicle by matching the current view with those in a set of views that has been previously obtained. The Robustness of matching to large view changes is, therefore, the key to reliable localization, and many approaches have been proposed. Such an image-to-image matching also appears in the context of loop closure detection for SLAM [23, 24] or in visual place recognition [25].

Local image descriptors such as SIFT [26] and SURF [27] are sometimes used for characterizing an image with, for example, the Bag-of-Visual-Words model [28]. This visual words-based approach is popular in outdoor localization [29, 4]. FAB-Map [3, 30] improves this approach by constructing and using a generative model of visual words based on their probabilistic relationships. Local image descriptors are, however, sometimes weak to large changes of weather and seasons [4].

Global image features such as GIST are also used for place recognition [31]. Combination of GIST with saliency-based landmarks is also proposed [32, 33] for localization in a topological map. However, since global image features still cannot cope with large view changes, an image database must be constructed by training under various lighting conditions [5].

Enhancing direct image-to-image matching is another approach, examples of which include a robust template matching [8], an SVM learning of objects with varying views [7, 6], and an illumination invariant image transformation [34]. Among those, the SeqSLAM method [9] achieves highly robust localization based on a robust image sequence matching. We extend this SeqSLAM for our view-based robot localization system.

3. View-based Localization by SeqSLAM and Its Improvements

3.1. SeqSLAM

SeqSLAM [9] is a view-based localization method which compares a reference image sequence with an input sequence for robust matching. It achieves highly accurate visual recognition by calculating the similarity between the reference and the input *image sequences* to estimate the best matching location instead of single image comparison. For robust estimation in an extreme lighting condition between training and testing time, it applies *local contrast enhancement* as follows.

Let $\boldsymbol{D}$ be the vector of the differences between an input image and the images in the reference sequence, which is considered to cover the possible range of reference images for the input image. Each element D_i in $\boldsymbol{D}$ is normalized by:

$$\hat{D}_i = \left(D_i - \bar{D}_l\right)/\sigma_l, \tag{1}$$

where D_l and σ_l are the mean and the standard deviation of $\boldsymbol{D}$. By this enhancement, even if an input image is largely different from the reference images and all of the difference values become large due to significant illumination change, the difference for the true correspondence is expected to be sufficiently small compared to the others.

These enhanced vectors are compiled for d_s+1 frames into a matrix $\boldsymbol{M}$ which has the reference and the input image sequence in the row and the column, respectively:

$$\boldsymbol{M} = \left[\hat{D}^{T-d_s}, \hat{D}^{T-d_s+1}, \ldots, \hat{D}^{T}\right] \tag{2}$$

An example matrix is shown in Fig. 4.

Then, assuming a constant velocity during the sequence, a line is searched for which minimizes the following total difference S:

$$S = \sum_{t=T-d_s}^{T} D_k^t, \quad (3)$$

$$k = s + V(d_s - T + t), \quad (4)$$

where V is the gradient of the line (or a relative velocity in input and reference acquisition) and k is the index of the corresponding image in the reference sequence for the input image at time t.

SeqSLAM exhibited great performances against drastic view changes, at least for road sequence images. There are, however, room for improvements when applied to mobile robot navigation. The following subsections explain our improvements.

3.2. SeqSLAM++ : Extension of the SeqSLAM for mobile robot navigation

3.2.1. Robust image matching and motion determination

The original SeqSLAM uses intensity values normalized within a small window as the feature for image matching. The method is simple and fast, however, is not robust to image deformations. In addition, it is not able to deal with large view direction changes due to the robot's motion. We therefore adopt two improvements: HOG feature matching and the use of a wide angle camera.

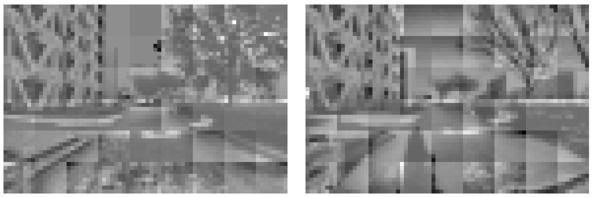
HOG feature matching: HOG feature [10] is a histogram of edges in a local region and suitable for representing shape information. The training images have 90° FOV (Field Of View), and the cell size for calculating HOG is determined so that each cell has 5° FOV. In addition, the number of bins is set as 9 and the block consists of 2×2 cells.

Fig. 2 shows an example of SeqSLAM's normalization and HOG calculation. Compared with the normalization method, HOG can extract an appearance feature invariant to lighting conditions and small deformations thanks to the discretized gradient representation and block normalization. The dissimilarity between images is obtained by calculating a normal SAD (Sum of Absolute Differences) in the proposed SeqSLAM++.

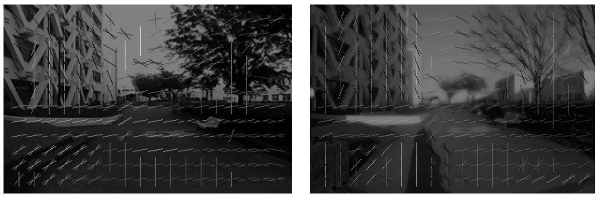
Moving direction determination to deal with a variety of robot motion: Mobile robots changes their moving directions frequently not only for moving toward a destination but also when avoiding collisions with people and obstacles. Since each image in a view sequence captures a scene in a specific direction, it is very likely to have a different orientation during navigation, thereby degrading the view-based localization performance.

Morita and Miura [11] used an omnidirectional camera to cope with this problem. We also take a similar approach using wide-angle images that have about 180° horizontal FOV as input images. A training image is scanned horizontally on the wide-angle input image within ±45° range at 5° interval, and choose the minimum distance position, which is then used for determining the subgoal direction (i.e., the direction for the robot to move). Fig. 3 shows an example of the direction determination.

(a) Example scenes.



(b) Image normalization results by SeqSLAM.



(c) Extracted HOG features.

Figure 2: Feature representation for image matching.

(a) Training image.

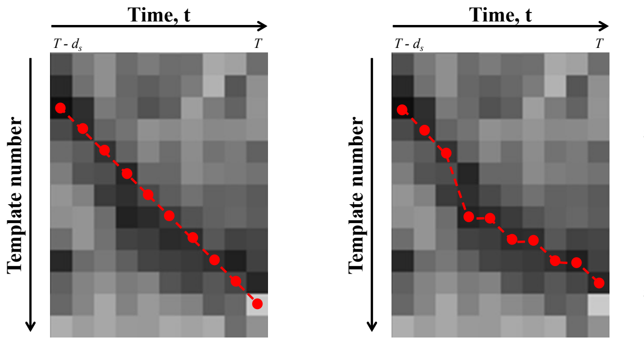
(b) Input image with the selected direction (red box).

Figure 3: Selection of moving direction.

3.2.2. Adaptive and efficient sequence matching against robot motion variations

The original SeqSLAM assumes a constant speed during the acquisition of training and input image sequences; the matrix is searched for the best line which minimizes the total difference. This assumption is sometimes violated in the case of mobile robots because they need to adjust their speed adaptively to the surrounding situation for, for example, avoiding a collision and/or not scaring people by getting too close. We therefore use a DP to cope with such speed variations during image acquisition. We also effectively utilize the history of movement to increase reliability and reduce the calculation cost.

DP Matching: DP (Dynamic Programming) matching [12] is a tool for calculating a match between two data sequences at non-constant intervals between data. Fig. 4 shows an example of DP matching for image sequences with non-constant robot motions; a linear matching is not suitable for this case. We set a limitation on the speed difference between the training and the navigation phase and apply the DP matching for obtaining the best matched image pairs with an evaluation. In addition, unlike SeqSLAM, we use the latest frame as a representative image of a sequence so that the current location is estimated on-line.



(a) line-based matching is not optimal. (b) DP matching can find the best matches.

Figure 4: DP matching between training and navigational image sequences. In the matrix, darker elements have smaller differences.

Markov localization: Mobile robot localization often uses a movement history, which is effective to limit the possible robot positions in prediction. Miura and Yamamoto [7] adopted a Markov localization strategy in a view-based localization. In [7], a discrete set of locations are provided for localization and a probabilistic model of transitions between locations was used in the prediction step. This can reduce not only localization failures but also the calculation cost with a limited number of sequence matches.

The Markov localization here is formulated as follows:

$$\hat{Bel}(l) \leftarrow \sum_{l'} P_m(l|l')Bel(l'), \tag{5}$$

$$Bel(l) \leftarrow \alpha P_o(s|l)\hat{Bel}(l), \tag{6}$$

$$P_o(s|l) = \frac{S_{min}}{S_l}, \tag{7}$$

where $P_m(l|l')$ denotes the transition probability from frame l' to l, $Bel(l)$ the belief of the robot being location l, $P_o(s|l)$ the likelihood of location l with sensing s, which is calculated by the minimum matching score of the DP matching divided by the score for location l.

The state transition model $P_m(l|l')$ is determined by considering the image acquisition interval and the robot's motion patterns. Currently, the training images are taken at about one meter interval and the robot takes one image per two seconds with moving at $1\,m/s$. Since the robot speed changes frequently due to various reasons such as collision avoidance and turning motions, we use a transition model in which the robot may move to locations corresponding to one of the current and the three subsequent location with equal probabilities.

4. Local Path Planning using View-Based Localization Results

The proposed SeqSLAM++ localizes the robot position and provides the direction to move. That information by itself is, however, not enough for guiding an actual robot safely. We therefore have implemented a local navigation system which includes local mapping, subgoal selection, and path planning.

4.1. Local mapping

A 3D LiDAR is used to create local 2D maps, which is for finding free spaces. We detect obstacles in two ways. One is to use a height map which detects regions with relatively high obstacles. We use a grid coordinate at $10\,cm$ intervals for representing height maps. The pose of the range sensors relative to the ground plane is estimated by fitting a plane to the data points in the region in front of the robot.

The other way is to find low steps. Since it is sometimes difficult to find such steps only from the height due to a limited ranging accuracy and the error in ground plane estimation, we examine the differentiation of height data. A region with a large height difference with a certain number of data points is considered to be an obstacle region (i.e., A low step). Fig. 5(b) shows an example of obstacle detection in a real scene as shown in Fig. 5(a). The obstacles including low steps in the surrounding environment are successfully detected.

The point cloud of the detected obstacle is periodically fed to GMapping[16] that is a well-known SLAM utilizing the Rao-Blackwellized particle filter[17] to generate a local map at each position (Fig. 5(c)).

4.2. Subgoal selection and path planning

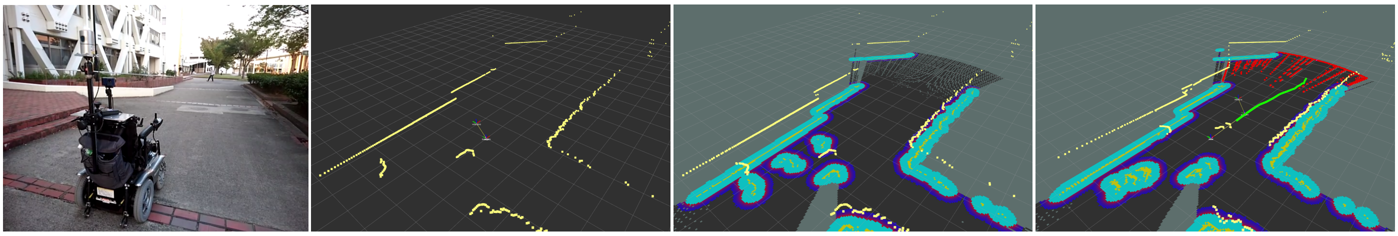
The direction to move is suggested by the proposed SeqSLAM++, however, it is not always possible to move in that direction due to obstacles. Since the free spaces are recognized only in the local map, we need to set a subgoal in the local map which is safe and leads the robot to the destination. We here adopt the concept of *frontier* which is often used in exploration planning in an unknown space [13]. A frontier point is a point which is free and adjacent to an unknown point. Such a point is either inside the local map or on the edge of the map. All frontier points in front of the robot are detected as candidates for the subgoal by examining each free cell in the local map whether it has any unknown points in the 8-neighbor cells around it. The most appropriate frontier point is then selected as the subgoal which has the minimum orientational difference with the suggested moving direction. Fig. 5(d) shows an example selection of subgoal (frontier).

Once the subgoal is set, the path toward it is generated. We use the A* and the Dynamic Window Approach (DWA) algorithm for global and local path planning, respectively[14, 15]. The SeqSLAM++ executes the view-based global localization at a certain interval, and a new subgoal is generated one after another according to the localization and moving direction estimation results.

5. Experiments

5.1. Experimental setup

Fig. 6 shows the mobile robot used in this research. It is based on an electric wheelchair (Patrafour by Toyota Motor East Japan Inc.) equipped with a wide-angle camera (Omni60 by Occam Vision Group) for view-based localizations, and 3D LiDAR(s) (Two FX-8x by Nippon Signal Co. or HDL-32e by Velodyne LiDAR, Inc.) for local mapping and finding free



(a) Scene. (b) Obstacle detection. (c) Local mapping and obstacle inflation. (d) Frontier detection and path planning.

Figure 5: Obstacle detection and subgoal selection : In a scene (a), the mobile robot detects obstacles based on a height map and projects them onto a 2D grid map (b). Feeding the obstacles into GMapping periodically, the robot builds a local 2D map at each position and finds the collision-free space (c) by inflating it. A subgoal is determined by selecting the *frontier* point that has the minimum orientation difference with the moving direction suggested by SeqSLAM++, then the path toward it (d) is generated with A* and DWA.

Figure 6: Our robot equipped with a wide-angle camera and a 3D LiDAR.

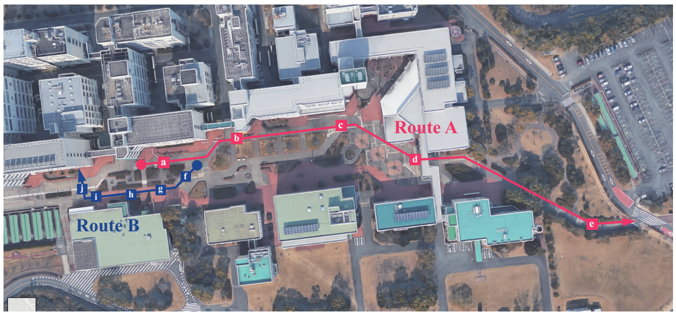


Figure 7: Experiment environment and routes for the navigation performance evaluation.

Table 1: Summaries of the training and testing image sets.

	Camera	Date and weather	# of images	Robot motion
Training	90° FOV	March 5, 2015, Fine	259	Smooth
Test 1	180° FOV	July 11, 2015, Cloudy	263	Smooth
Test 2	180° FOV	July 11, 2015, Cloudy	282	Zig-zag

spaces to generate subgoals. The HDL-32e has 360° horizontal FOV and measures the structure of the surrounding environment. Although the Omni60 also has 360° horizontal FOV, the frontal $90° and 180°$ parts are used for capturing the training and input images, respectively.

The following experiments were carried out in the campus of Toyohashi University of Technology. As shown in Fig. 7, two routes are defined: Route A is about 300 m long and relatively straight while Route B has some curves where the mobile robot needs to make a sharp turn to follow it.

5.2. *Off-line localization*

In this experiment, we manually moved the robot making it follow route A and acquired a set of training images and two sets of test images. The training and the first test image sets were taken while the robot moved along the route. On the other hand, the second test image set was captured as the robot moved in a zig-zag so that the direction of the robot changes largely from position to position. The image sets are shown in Fig. 8 and the detail is summarized in Table 1.

We compared the localization performance of the SeqSLAM and the proposed SeqSLAM++ for the test image sequences. Fig. 9 shows the localization results. Note that since the original SeqSLAM exhibited quite a low performance for the second test image sequence due to a large variation of the robot's heading, we additionally implemented the same horizontal scanning in the SeqSLAM to find the best matched position in the wide image. Both comparison results show that the proposed method exhibits a much better performance.

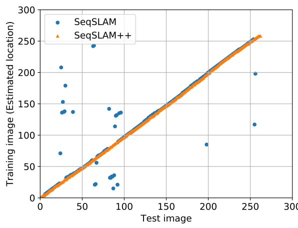
Fig. 10 shows the evaluation of localization accuracy. The ground truth is determined by manually comparing the training and test images. When an input image is judged to be located between two consecutive training images, the true position is set in the middle of the training images. The maximum frame difference by the proposed method is two for most of frames, meaning the maximum localization error is about 2 m because the training images are acquired at about one meter interval. Table 2 summarizes the performance in terms of *localization success rate* and *direction selection success rate*. Localization is considered success when the difference is within two frames, while the direction is considered correctly selected when the directional difference is less than 5°. Fig. 11 shows a scene where the proposed and the SeqSLAM with horizontal scanning suggest different moving directions. Since SeqSLAM does a direct comparison of (normalized) pixel values, it is sometimes

(a) Training image sequence along the route A.

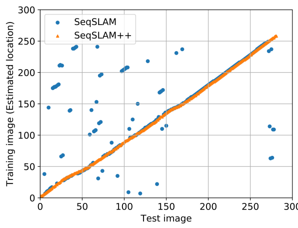
(b) Input image sequence with smooth motion (test1).

(c) Input image sequence with zig-zag motion (test2).

Figure 8: Image sets captured along the route A in Fig. 7 for off-line localization experiment.

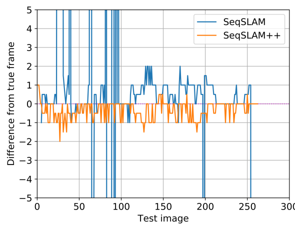


(a) Test image sequence 1 (Smooth).

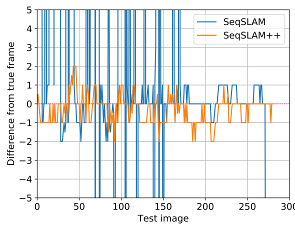


(b) Test image sequence 2 (Zig-zag).

Figure 9: Localization results.



(a) Test image sequence 1 (Smooth).



(b) Test image sequence 2 (Zig-zag).

Figure 10: Localization accuracy.

Table 2: Quantitative evaluation results.

Test image	Method	Localization success rate (%)	Direction selection success rate (%)
Test 1	SeqSLAM	86.1	-
	SeqSLAM++	99.6	99.6
Test 2	SeqSLAM	74.2	63.1
	SeqSLAM++	95.8	95.8

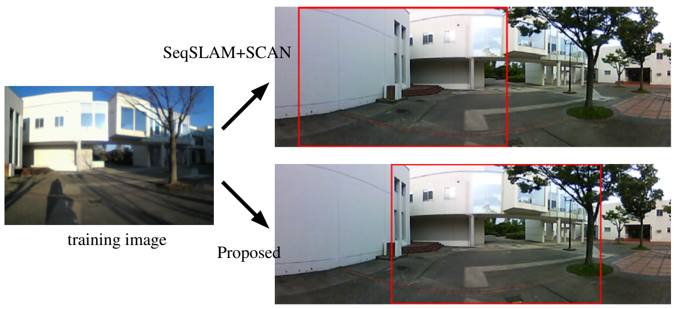


Figure 11: Comparison in selecting the moving direction.

weak to scenes with less textures as shown in the figure.

5.3. Autonomous navigation

Next, we conducted autonomous navigation experiments in the routes shown in Fig. 7. In the same way as in Sec. 5.2, training datasets were captured manually by moving the robot to follow the routes. On the other hand, the input images were taken in real time while the mobile robot was controlling itself according to the outputs of SeqSLAM++. Fig. 12 shows snapshots of the runs, and the names of the locations correspond to symbols in Fig. 7.

As shown in Fig. 12, the robot successfully completed the autonomous navigation in each routes localizing its position based on the view sequence. Additionally, some parts of the actual trajectories during the navigation were measured to evaluate the accuracy of the robot motion with the LiDAR-based localization technique that was capable of estimating the robot position correctly by comparing the global 3D map created in advance and the local range data with Normal Distribution Transformation (NDT) [35].

As shown in Fig. 13, the mobile robot follows the training routes properly and the proposed SeqSLAM++ combined with local planning works on-line with high reliability. When it was difficult to exactly follow the path and when the robot slightly went off the target route due to static/dynamic obstacles or the structure of the surrounding environment, local path planning provided appropriate paths to keep the robot traveling in the right direction given by SeqSLAM++.

Location a.

Location b.

Location c.

Location d.

Location e.

(a) Route A.

Location f.

Location g.

Location h.

Location i.

Location j.

(b) Route B.

Figure 12: Navigation experiments. Left:Robot in motion. Center:Estimated location (i.e., best matched training image). Right:Input image and selected moving direction.

6. Conclusions and Future Work

In this paper, we proposed a new view-based localization and navigation system *SeqSLAM++* by extending the original SeqSLAM to deal with several issues in mobile robot navigation. The SeqSLAM++ is capable of localizing the robot robustly even when the input images have deformations or view direction changes from the training images due to the robot motion by taking advantage of HOG feature and DP dissimilarity calculation in the sequence matching, the horizontal scanning in the wider input image, and the probabilistic model based on Markov localization strategy. The reliability has been shown in the off-line evaluations in an outdoor environment, and achieved 95.8% localization success rate against the dataset with large view direction changes.

In addition, we have also developed an autonomous system that can navigate itself given an image sequence of the target route to follow. Combined with the local mapping and path planning algorithm, the mobile robot successfully reached the destinations tracing subgoals generated by SeqSLAM++ according to the sequence matching results.

The entire system has been validated in just two routes in the campus of Toyohashi University of Technology. To examine the ability and limitation of the proposed SeqSLAM++, it is necessary to evaluate the system in a wider variety of scenes, that is, more variations in weather, season, surrounding objects (buildings or forests), and so on.

At the same time, toward an inexpensive and easily applicable navigation system, it is essential to develop an alternative local mapping algorithm with visual SLAM techniques. In the current system, geometric information around the mobile robot is obtained with an expensive 3D LiDAR. Instead of using that kind of range sensor, we will build a comprehensive system that performs the global-level localization with SeqSLAM++ utilizing local-level motion planning with 3D reconstruction techniques for a mobile robot equipped only with digital cameras.

Acknowledgment

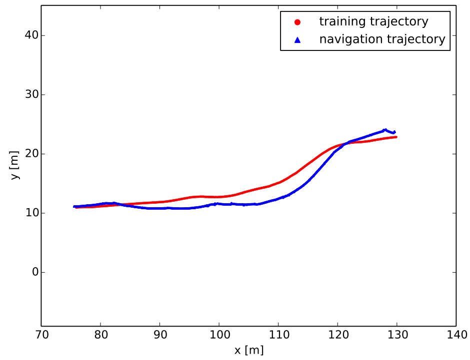
This work is in part supported by JSPS KAKENHI Grant Number 25280093.

References

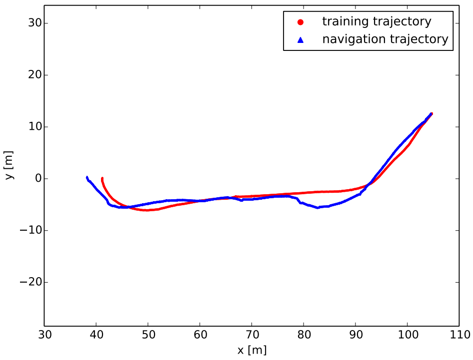
[1] Y. Matsumoto, M. Inaba, and H. Inoue. Visual Navigation using View-Sequenced Route Representation. In *Proceedings of 1996 IEEE Int. Conf. on Robotics and Automation*, pp. 83–88, 1996.

[2] H. Katsura, J. Miura, M. Hild, and Y. Shirai. A View-Based Outdoor Navigation Using Object Recognition Robust to Changes of Weather and Seasons. In *Proceedings of 2003 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2974–2979, 2003.

[3] 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.



(a) Route A (from the starting point to the point b in Fig. 7).



(b) Route B.

Figure 13: Trajectories in the training and navigation driving.

[4] C. Valgren and A. Lilienthal. SIFT, SURF and Seasons: Long-term Outdoor Localization Using Local Features. In *Proceedings of European Conf. on Mobile Robots*, pp. 253–258, 2007.

[5] D.M. Bradley, R. Patel, N. Vandapel, and S.M. Thayer. Real-Time Image-Based Topological Localization in Large Outdoor Environments. In *Proceedings of the 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 3062–3069, 2005.

[6] H. Morita, M. Hild, J. Miura, and Y. Shirai. View-Based Localization in Outdoor Environments Based on Support Vector Learning. In *Proceedings of 2005 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 3083–3088, 2005.

[7] 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.

[8] Y. Yamagi, J. Ido, K. Takemura, Y. Matsumoto, J. Takamatsu, and T. Ogasawara. View-sequence Based Indoor/Outdoor Navigation Robust to Illumination Changes. In *Proceedings of 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 1229–1234, 2009.

[9] M.J. Milford and G.F. Wyeth. SeqSLAM: Visual Route-Based Navigation for Sunny Summer Days and Stormy Winter Nights. In *Proceedings 2012 IEEE Int. Conf. on Robotics and Automation*, 2012.

[10] N. Dalal and B. Briggs. Histograms of Oriented Gradients for Human Detection. In *Proceedings of 2005 IEEE Conf. on Computer Vision and Patttern Recognition*, pp. 886–893, 2005.

[11] H. Morita, M. Hild, J. Miura, and Y. Shirai. Panoramic View-Based Navigation in Outdoor Environments Based on Support Vector Learning. In *Proceedings of 2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2302–2307, 2006.

[12] Y. Ohta and H. Yamada. Pattern Matching Using Dynamic Programming. *Information Processing*, Vol. 30, No. 9, pp. 1058–1066, 1989.

[13] B. Yamauchi. A Frontier-Based Approach for Autonomous Navigation. In *Proceedings of the 1997 IEEE Int. Conf. on Computational Intelligence in Robotics and Automation*, pp. 146–151, 1997.

[14] E.H. Peter, N.J. Nils, R. Bertram. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, pp. 100–107, 1968.

[15] D. Fox, W. Burgard, and S. Thrun. The Dynamic Window Approach to Collision Avoidance. *IEEE Robotics and Automation Magazine*, Vol. 4, No. 1, 1997.

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

[17] K. Murphy, S. Russell. Rao-blackwellised particle filtering for dynamic bayesian networks, Sequential Monte Carlo Methods in Practice. *Springer*, 2001.

[18] R. Thrapp, C. Westbrook, and D. Subramanian. Robust Localization Algorithms for an Autonomous Campus Tour Guide. In *Proceedings of 2001 IEEE Int. Conf. on Robotics and Automation*, pp. 2065–2071, 2001.

[19] K. Ohno, T. Tsubouchi, B. Shigematsu, and S. Yuta. Differential GPS and Odometry-based Outdoor Navigation of a Mobile Robot. *Advanced Robotics*, Vol. 18, No. 6, pp. 611–635, 2004.

[20] S. Thrun and M. Montemerlo. The GraphSLAM Algorithm with Applications to Large-Scale Mapping of Urban Structures. *Int. J. of Robotics Research*, Vol. 25, No. 6, pp. 403–429, 2006.

[21] D.M. Cole and P.M. Newman. Using Laser Range Data for 3D SLAM in Outdoor Environments. In *Proceedings of 2006 IEEE Int. Conf. on Robotics and Automation*, pp. 1556–1563, 2006.

[22] A. Nüchter, K. Lingemann, J. Hertzberg, and H. Surmann. 6D SLAM – 3D Mapping Outdoor Environments. *J. of Field Robotics*, Vol. 24, No. 8–9, pp. 699–722, 2007.

[23] K.L. Ho and P. Newman. Detecting Loop Closure with Scene Sequences. *Int. J. of Computer Vision*, Vol. 74, No. 3, pp. 261–286, 2007.

[24] M. Labbé and F. Michaud. Appearance-based loop closure detection for online large-scale and long-term operation. *IEEE Trans. on Robotics*, Vol. 29, No. 3, pp. 734–745, 2013.

[25] S. Lowry, N. Sünderhauf, P. Newman, J.J. Leonard, D. Cox, P. Corke, and M.J. Milford. Visual Place Recognition: A Survey. *IEEE Trans. on Robotics*, Vol. 32, No. 1, pp. 1–19, 2016.

[26] D.G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. of Computer Vision*, Vol. 60, No. 2, pp. 91–110, 2004.

[27] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded Up Robust Features. In *Proceedings of the 9th European Conf. on Computer Vision*, pp. 404–417, 2006.

[28] C. Csurka, C.R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual Categorization with Bags of Keypoints. In *Proceedings of ECCV-2004 Workshop on Statistical Learning in Computer Vision*, pp. 1–22, 2004.

[29] C. Silpa-Anan and R. Hartley. Visual Localization and Loop-Back Detection with a High Resolution Omnidirectional Camera. In *Proceedings of the 6th Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras (OMNIVIS 2005)*, 2005.

[30] M. Cummins and P. Newman. Appearance-Only SLAM at Large Scale with FAB-MAP 2.0. *Int. J. of Robotics Research*, Vol. 30, No. 9, pp. 1100–1123, 2010.

[31] C. Siagian and L. Itti. Rapid Biologically-inspired Scene Classification using Features Shared with Visual Attention. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 2, pp. 300–312, 2009.

[32] C. Siagian and L. Itti. Biologically Inspired Mobile Robot Vision Localization. *IEEE Trans. on Robotics*, Vol. 25, No. 4, pp. 861–873, 2009.

[33] C.-K. Chang, C. Siagian, and L. Itti. Mobile Robot Vision Navigation and Localization Using Gist and Saliency. In *Proceedings of 2010 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 4147–4154, 2010.

[34] W. Maddern, A.D. Stewart, C. McManus, B. Upcroft, W. Churchill, and P. Newman. Illumination Invariant Imaging: Applications in Robust Vision-based Localisation, Mapping and Classification for Autonomous Vehicle. In *Proceedings of 2014 IEEE Int. Conf. on Robotics and Automation*, 2014.

[35] M. Magnusson, A. Nuchter, C. Lorken, A.J. Lilienthal, and J. Hertzberg. Evaluation of 3D registration reliability and speed - a comparison of ICP and NDT. In *IEEE International Conference on Robotics and Automation*, 2009.