2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

SLAMER: Simultaneous Localization and Map-Assisted Environment Recognition

Naoki Akai[1,2]

***Abstract*— This paper presents a simultaneous localization and map-assisted environment recognition (SLAMER) method. Mobile robots usually have an environment map and environment information can be assigned to the map. Important information such as no entry zone can be predicted from the map if localization has succeeded. However, this prediction is failed when localization does not work. Uncertainty of pose estimate must be considered for robust-map-based environmental object prediction. Robots also have external sensors and can recognize environmental object; however, sensor-based recognition of course contain uncertainty. SLAMER fuses map-based prediction and sensor-based recognition while coping with these uncertainties and achieves accurate localization and environment recognition. In this paper, we demonstrate LiDAR-based implementation of SLAMER in two cases. In the first case, we use the SemanticKITTI dataset and show that SLAMER achieves accurate estimate more than traditional methods. In the second case, we use an indoor mobile robot and show that unmeasurable environmental objects such as open doors and no entry lines can be recognized.**

I. Introduction

Autonomous mobile robots usually have an environment map and any information can be assigned to the map. The assigned environmental information can be predicted if localization has succeeded. This map-based prediction enables robots to effectively recognize important information such as no entry zone [1]. However, this prediction is failed when localization does not work. Uncertainty of pose estimate must be considered for robust-map-based environmental object prediction. Robots also have external sensors and can recognize environmental object; however, sensor-based recognition also contains uncertainty. This paper presents a simultaneous localization and map-assisted environment recognition (SLAMER) method to simultaneously cope with these uncertainties.

The graphical model of SLAMER is illustrated in the top of Fig. 1. The white and gray nodes represent hidden and observable variables and the arrows represent dependencies where the tip variables depend on the root variables. A pose of a robot, $\mathbf{x}$, and true environmental object classes, $\mathbf{c}$, are treated as the hidden variables, and control input, $\mathbf{u}$, sensor measurement, $\mathbf{z}$, semantic map, $\mathbf{m}$, environmental object recognition results, $\hat{\mathbf{c}}$, and hyperparameters of the recognition method, Θ, are treated as the observable variables. These variables are detailed in Section III-A.

In the general localization model presented in [2], $\hat{\mathbf{c}}$ and $\mathbf{c}$ do not exist. In the SLAMER model, these variables are introduced as the observable and hidden variables, and we assume that the recognition results depend on the true classes. The true classes can be estimated based on the recognition results by the relationship. In addition, we assume that the true classes depend on the pose and the semantic map. The pose and map information can be used to predict the true classes by the relationship. Owing to these relationships, the true object classes can be estimated while respecting to both map-based prediction and sensor-based recognition, and these results are fused with the Bayes theorem. Consequently, uncertainties in localization and environment recognition can be simultaneously considered. The bottom figures of Fig. 1 shows estimate examples by 2D-LiDAR-based SLAMER. An important advantage of SLAMER is that objects which do not provide better influence to localization such as no entry line (red), glass door (purple), and open door (green) can be treated in its framework.

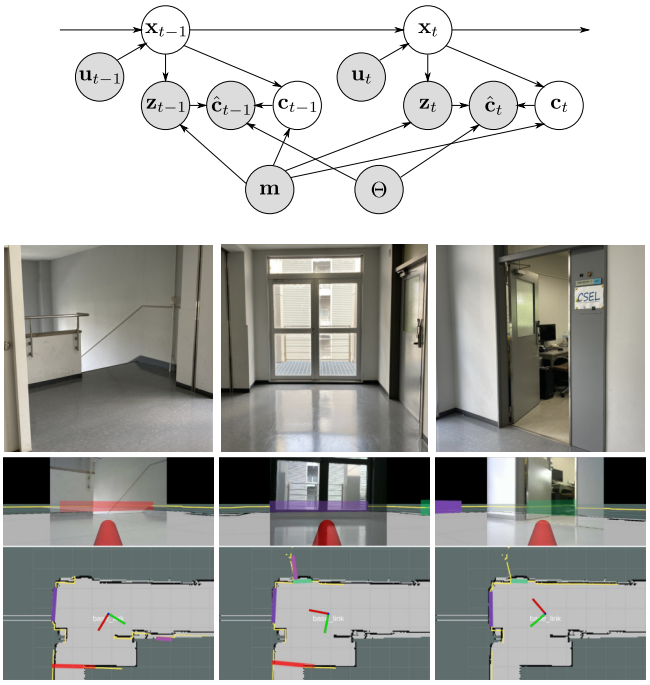

Fig. 1. Top: The graphical model of SLAMER. Bottom: Estimate examples by 2D LiDAR-based SLAMER. No entry line (red), glass door (purple), and open door (green) are recognized utilizing the map assist.

In this paper, we demonstrate LiDAR-based implementation of SLAMER with two cases; urban driving car equipped with 3D LiDAR and indoor mobile robot equipped with 2D LiDAR. In the urban driving case, we use the SemanticKITTI

[1]Naoki Akai is with the Graduate School of Engineering, Nagoya University, Nagoya 464-8603, Japan akai@nagoya-u.jp
[2]Naoki Akai is with LOCT Co., Ltd., Nagoya 466-0805, Japan

dataset [3] and show quantitative results including comparison. The comparison show that SLAMER outperforms traditional methods; however, it cannot outperform the class prediction model (CPM), that is presented in [4] for performing localization with semantics, in terms of localization accuracy. This paper also discusses the advantage and disadvantage of SLAMER while comparing with CPM. In the indoor mobile robot case, we use our own experimental platform to show qualitative results as shown in the bottom of Fig. 1. Through the demonstrations, we show that SLAMER improves both localization and environment recognition accuracy. The contribution of this paper is twofold.

- Proposing the general framework for simultaneous localization and map-assisted environment recognition
- Discussing advantage and disadvantage of SLAMER while considering relationship with CPM

II. Related work

Recent evolution of machine learning algorithms enables us to obtain accurate semantic information from camera images [5] and/or 3D point clouds [6]. Owing to the breakthroughs, the use of semantics in localization and SLAM becomes popular in recent.

A semantic-aware visual localization method is presented in [7]. In [7], the semantics-assisted measurement model is presented. A deep-learning-based visual localization method with semantics presented in [8]. The network used in the method takes incomplete semantic observations and predicts completed semantic subvolumes. The method also uses joint geometric and semantic matching that confirms whether voxels are matched well in term of both geometric and semantics. A semantic-aware visual SLAM method is presented in [9], [10]. In [9], dynamic and static keypoints can be separated with the semantics. In [10], the geometric error term that enables to use object detection systems such as Faster R-CNN [11] as sensors in SLAM is presented. A semantic-aware localization method using a LiDAR is presented in [12], [13]. In these works, the semantics is utilized in the cost function for point cloud registration to mitigate influence of outliers. A semantic-aware SLAM method using a LiDAR is presented in [14] and it also uses the semantics in the cost function for registration. These methods use the semantics for improving localization and mapping accuracy; however, uncertainty in semantics recognition is not considered because these uses are similar to robust estimation such as M-estimation [15]. In addition, map-assisted environment recognition is not focused on.

Parkison *et al*. [16] presented semantic ICP scan matching with the Expectation-Maximization (EM) algorithm. Through the EM algorithm, ambiguity of the semantics can be coped with. The similar approaches that uses Markov random fields or conditional random fields to cope with outliers in scan matching can be seen at [17], [18]. Bowman *et al*. [19] presented the probabilistic data association method for semantic SLAM using the k-best assignment enumeration. The k-best assignment enables to compute marginal assignment probabilities for each measurement landmark pair. Atanasov *et al*. [20] the localization method using the semantic observations. They proposed the sensor model encodes the semantics via random finite sets and it realizes a unified treatment of miss detection, false alarms, and data association. They also proposed the efficient likelihood calculation method using the matrix permanent. These methods consider uncertainty in semantics recognition. Our proposal extends these approaches to a simultaneous environment recognition approach which utilizes the semantic map, i.e., semantics recognition results are improved while performing localization.

Our proposal performs simultaneous localization and environmental recognition. A similar approach that simultaneously updates an environment map while performing localization in dynamic environments is presented in [21]. This probabilistic simultaneous estimation is typically implemented based on Rao-Blackwellized particle filter, e.g., FastSLAM 2.0 [22], and this implementation requires large memory cost since each particle has to have a map. However, the proposed method does not require large memory cost since the particles do not have a map. Wang *et al*. [23] presented the simultaneous localization, mapping, and moving object tracking method. This simultaneous estimation is also similar to our proposal, but our focus is different from their focus. SLAMER performs localization while doing map-assisted environment recognition.

The most related work to SLAMER is our previous work [4]. In [4], localization using semantic information is presented. In the method, deep-learning-based semantic segmentation is used. The localization method can cope with uncertainty of semantic segmentation by the class prediction model (CPM). CPM models probabilistic distribution over the semantic segmentation results based on the off-line test. Owing to the use of CPM, localization accuracy and robustness to inaccurate object recognition can be improved. However, CPM can only be used for likelihood calculation for localization, that is, spatial environmental information, such as unmeasurable walls and no entry lines which might not be used for localization, cannot be handled. SLAMER provides an extended framework that can be used for recognition of such spatial objects.

III. Proposed method

A. Target problem and variable definition

In this work, we focus on mobile robot localization and environment recognition with LiDAR and semantic map. We denote a robot pose as $\mathbf{x}$. The robot is equipped with an inertial measurement system (INS) and a LiDAR and these measurements are denoted by $\mathbf{u}$ and $\mathbf{z}$. The semantic map is represented by a grid or voxel map, $\mathbf{m} = (m_1, ..., m_M)$, where M is number of cells and $m_i \in \mathcal{L}$ is a ith cell's environmental object label included in the environmental object list $\mathcal{L}$.

In SLAMER, an environment recognition method is used. The recognition results and the hyperparameters of the method are denoted by $\hat{\mathbf{c}}$ and Θ. The recognition method can be implemented with any methods. In this work, we assume

that the recognition method outputs probability over the environmental object classes, i.e., $\hat{\mathbf{c}} = (\hat{\mathbf{c}}^{[1]}, ..., \hat{\mathbf{c}}^{[{}^{\mathrm{e}}K]})$, $\hat{\mathbf{c}}^{[k]} = (\hat{c}^{[k,1]}, ..., \hat{c}^{[k,L]})$, $0 \leq \hat{c}^{[k,l]} \leq 1$, and $\sum_{l \in \mathcal{L}} \hat{c}^{[k,l]} = 1$, where ${}^{\mathrm{e}}K$ and L are numbers of recognized environmental objects and object classes. We further assume that the true classes, $\mathbf{c}$, can be estimated using the recognition results, where $\mathbf{c} = (\mathbf{c}^{[1]}, ..., \mathbf{c}^{[{}^{\mathrm{e}}K]})$, $\mathbf{c}^{[k]} = (c^{[k,1]}, ..., c^{[k,L]})$, $c^{[k,l]} \in \{0, 1\}$, and $\sum_{l \in \mathcal{L}} c^{[k,l]} = 1$. It should be noted that the hyperparameters depend on implementation of the recognition method.

B. Formulation

The graphical model of SLAMER is shown in the top of Fig. 1. The objective of SLAMER is to estimate the joint posterior distribution shown in Eq. (1).

$$p(\mathbf{x}_t, \mathbf{c}_t | \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \hat{\mathbf{c}}_{1:t}, \mathbf{m}, \Theta), \tag{1}$$

where t and $1:t$ represent current and time sequence data. Eq. (1) can be decomposed using the multiply theorem.

$$p(\mathbf{x}_t | \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \hat{\mathbf{c}}_{1:t}, \mathbf{m}, \Theta) p(\mathbf{c}_t | \mathbf{x}_t, \mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \hat{\mathbf{c}}_{1:t}, \mathbf{m}, \Theta).$$

Then, we focus on how these two terms can be formulated.

The left term can be re-written using the Bayes and low of total probability theorems and D-separation [24].

$$\underbrace{\eta \sum_{\mathbf{c}_t} \left\{ p(\hat{\mathbf{c}} | \mathbf{c}_t, \mathbf{z}_t, \Theta) p(\mathbf{c}_t | \mathbf{x}_t, \mathbf{m}) \right\} p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{m})}_{\text{likelihood distribution}}$$
$$\underbrace{\int p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t) p(\mathbf{x}_{t-1} | \mathbf{u}_{1:t-1}, \mathbf{z}_{1:t-1}, \hat{\mathbf{c}}_{1:t-1}, \mathbf{m}, \Theta) \mathrm{d}\mathbf{x}_{t-1}}_{\text{predictive distribution}}, \tag{2}$$

where η is a normalization constant, $p(\hat{\mathbf{c}} | \mathbf{c}_t, \mathbf{z}_t, \Theta)$ is the environmental object recognition model, $p(\mathbf{c}_t | \mathbf{x}_t, \mathbf{m})$ is the prior distribution over the environmental object classes based on the semantic map, $p(\mathbf{z}_t | \mathbf{x}_t, \mathbf{m})$ is the measurement model, and $p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t)$ is the motion model. In the general localization model presented in [2], the measurement model is only used as the likelihood distribution; however, additional two models are used in the SLAMER's likelihood distribution. Because SLAMER contains these two distribution in the likelihood distribution, it can cope with uncertainties of environment recognition.

The right term can also be re-written using the Bayes and low of total probability theorems and D-separation.

$$\eta p(\hat{\mathbf{c}} | \mathbf{c}_t, \mathbf{z}_t, \Theta) p(\mathbf{c}_t | \mathbf{x}_t, \mathbf{m}) \tag{3}$$

Eq. (3) shows that map-based environment recognition is updated using the environmental object recognition model. As a result, robustness in environment recognition can be improved. In the next, implementation ways are detailed.

IV. Implementation

Our target is to estimate the joint posterior distribution shown in Eq. (1). To estimate the posterior, Rao-Blackwellized particle filter (RBPF) is used. Specifically, the distributions shown in Eqs. (2) and (3) are estimated with analytical and sampling-based methods, respectively. This section describes main processes to implement RBPF.

A. Motion model

We assume that the robot's motion model can be denoted as $\mathbf{x}_t = \mathbf{f}(\mathbf{x}_{t-1}, \mathbf{u}_t)$, where $\mathbf{f}(\cdot)$ is the motion model. ith particle pose, $\mathbf{x}_t^{[i]}$, is updated as follow.

$$\mathbf{x}_t^{[i]} = \mathbf{f}(\mathbf{x}_{t-1}^{[i]}, \mathbf{u}_t^{[i]}), \quad \mathbf{u}_t^{[i]} \sim \mathcal{N}(\mathbf{u}_t, \Sigma_{\mathrm{u}}),$$

where $\mathcal{N}(\mathbf{u}_t, \Sigma_{\mathrm{u}})$ is the normal distribution with mean, $\mathbf{u}_t$, and covairnace, Σ_{u}. In both the experiments conducted in this paper, the differential drive model is used.

B. Environmental object recognition

In this work, we tested SLAMER in two cases; SemanticKITTI dataset and indoor mobile robot cases. In these cases, we used different recognition methods. These are detailed in Sections V-B and VI-B, respectively. It should be noted that both the recognition methods output probability over the environment object classes, $\hat{\mathbf{c}}$, and it is used in the likelihood calculation and the update of the environmental object recognition results.

C. Likelihood calculation

ith particle's likelihood, $\omega_t^{[i]}$, is calculated as follow.

$$\omega_t^{[i]} \propto \prod_{k=1}^{{}^{\mathrm{e}}K} \left(\sum_{l \in \mathcal{L}} \left\{ p(\hat{\mathbf{c}}_t^{[k]} | c_t^{[k,l]}, \hat{\mathbf{z}}_t^{[k]}, \Theta) p(c_t^{[k,l]} | \mathbf{x}_t^{[i]}, \mathbf{m}) \right\} \right) \prod_{k=1}^{{}^{\mathrm{r}}K} p(\mathbf{z}_t^{[k]} | \mathbf{x}_t^{[i]}, \mathbf{m}),$$

where ${}^{\mathrm{e}}K$ and ${}^{\mathrm{r}}K$ are the numbers of the recognized objects and the LiDAR measurement used for localization. We assumed that the environmental object recognition results and sensor measurement are independent one another, and the probabilistic models included in the likelihood distribution shown in Eq. (2) can be decomposed as presented in [2].

The prior over the object classes is modeled using the normal distribution.

$$p(c_t^{[k,l]} | \mathbf{x}_t^{[i]}, \mathbf{m}) = \mathcal{N}(d_t^{[k,l]}; 0, \sigma_{\mathrm{d}}), \tag{4}$$

where $d_t^{[k,l]}$ is a representative distance from the recognized object to the closest obstacles existing in lth semantic map.

The environmental object recognition model is modeled using the Dirichlet distribution.

$$p(\hat{\mathbf{c}}_t^{[k]} | c_t^{[k,l]}, \hat{\mathbf{z}}_t^{[k]}, \Theta) = \mathrm{Dir}\left(\hat{\mathbf{c}}_t^{[k]}; \mathbf{a}(\hat{\mathbf{z}}_t^{[k]}, \Theta)\right) = \frac{\Gamma\left(\sum_{l \in \mathcal{L}} a^{[k,l]}\right)}{\prod_{l \in \mathcal{L}} \Gamma(a^{[k,l]})} \prod_{l \in \mathcal{L}} \left(\hat{c}_t^{[k,l]}\right)^{a^{[k,l]}-1}, \tag{5}$$

where $\mathbf{a}(\cdot) = (a^{[k,1]}, ..., a^{[k,L]})$, $a^{[k,l]} > 0$ is the hyperparameters and $\hat{\mathbf{z}}_t^{[k]}$ is the measurement used for recognizing kth environmental object. In this work, we determined the hyperparameters as follow.

$$a^{[k,l]} = \begin{cases} a_1 & \text{if } c_t^{[k,l]} = 1 \\ a_2 & \text{otherwise,} \end{cases} \tag{6}$$

where a_1 and a_2 are arbitrary constants. Concrete values of them are determined while respecting to performance of the recognition method.

The measurement model is modeled using the likelihood field model (LFM), $p_{\mathrm{lfm}}(\mathbf{z}_t^{[k]}|\mathbf{x}_t^{[i]}, \mathbf{m})$, [2]. It should be noted that the semantics is not considered in calculation of LFM because the SLAMER model does not assume that the environmental object recognition results do not depend on the semantic map.

D. Update environmental object recognition results

Environmental object recognition is performed based on the method described in Section IV-B. Then, Eqs. (4) and (5) are used to obtain the posterior over the true object classes. It should be noted that the pose $\mathbf{x}_t^{[i]}$ is replaced to that of the maximum likelihood particle in this update and this updated result is used as the final environment recognition results.

V. Experiments with dataset

In this section, we evaluate SLAMER of 3D LiDAR-based implementation on the SemanticKITTI dataset [3].

A. Setup

The SemanticKITTI dataset has vehicle trajectories and corresponding 3D point clouds. The ground truth object labels are assigned to each scan point. We first plotted the 3D point clouds obtained from the static objects such as buildings and roads according to the vehicle trajectory and built 3D maps. The method presented in [25] was used for building a distance field map which enables to efficiently get the closest distance from the obstacles. We then simulated noisy odometry measurements from the trajectories. In the experiment step, the noisy odometry was given to update the vehicle pose and accumulated errors were compensated by matching of the 3D point clouds with the map. The estimated poses were compared with the given trajectories and estimation accuracy was calculated.

500 particles were used to implement RBPF. The hyperparameters shown in Eq. (6) were experimentally set to $a_1 = 1.2$ and $a_2 = 1$.

B. Deep-learning-based environment recognition

We used the same environment recognition method presented in [4]. In [4], the SegNet [5]-based method was used. The depth and intensity maps are made from the 3D point cloud and these maps are fed to the network. The network infers probability over the object classes for each pixel and the output probabilities are treated as $\hat{\mathbf{c}}$. The probability is calculated via the softmax function.

C. Comparison methods for pose estimation

1) Likelihood field model (LFM): The general localization with LFM is formulated as the recursive Bayes filter.

$$p(\mathbf{x}_t|\mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{m}) = \eta \prod_{k=1}^{K} p_{\mathrm{lfm}}(\mathbf{z}_t^{[k]}|\mathbf{x}_t, \mathbf{m}) \int p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_t) p(\mathbf{x}_{t-1}|\mathbf{u}_{1:t-1}, \mathbf{z}_{1:t-1}, \mathbf{m}) \mathrm{d}\mathbf{x}_{t-1}. \tag{7}$$

2) Semantic likelihood field model (SLFM): We also implemented LFM with the estimated object classes to estimate Eq. (7). The likelihood distribution of Eq. (7) is replace to

$$\prod_{k=1}^{K} \prod_{l \in \mathcal{L}} p_{\mathrm{slfm}}(\mathbf{z}_t^{[k]}|\mathbf{x}_t, \mathbf{m}_l)^{\mathbb{1}\left(\text{if } \hat{c}^{[k,l]} \text{ is max in } \hat{\mathbf{c}}^{[k]}\right)},$$

where $\mathbb{1}(\cdot)$ is an indicator function which is equal to 1 when the condition within the bracket is true, and 0 otherwise, and $p_{\mathrm{slfm}}(\mathbf{z}_t^{[k]}|\mathbf{x}_t, \mathbf{m})$ is denoted as follow.

$$p_{\mathrm{slfm}}(\mathbf{z}_t^{[k]}|\mathbf{x}_t, \mathbf{m}_l) = \begin{cases} \frac{\lambda \exp\left(-\lambda r_t^{[k]}\right)}{1-\exp(-\lambda r_{\max})} & \text{if } l \in \text{unknown} \\ p_{\mathrm{lfm}}(\mathbf{z}_t^{[k]}|\mathbf{x}_t, \mathbf{m}_l) & \text{otherwise,} \end{cases}$$

where λ is the hyperparameter for the exponential distribution, and $\mathbf{m}_l$ is lth label's semantic map. We refer this method to the semantic likelihood field model (SLFM).

3) Class prediction model (CPM): The class prediction model (CPM), $p(\hat{\mathbf{c}}_t|\mathbf{x}_t, \mathbf{z}_t, \mathbf{m}, \Theta)$, is presented in [4] and it is used for estimating the posterior over the pose.

$$p(\mathbf{x}_t|\mathbf{u}_{1:t}, \mathbf{z}_{1:t}, \mathbf{m}, \Theta) = \eta p(\hat{\mathbf{c}}_t|\mathbf{x}_t, \mathbf{z}_t, \mathbf{m}, \Theta) \int p(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_t) p(\mathbf{x}_{t-1}|\mathbf{u}_{1:t-1}, \mathbf{z}_{1:t-1}, \hat{\mathbf{c}}_{1:t-1}\mathbf{m}, \Theta) \mathrm{d}\mathbf{x}_{t-1}.$$

CPM is also decomposed to ensure feasibility of calculation. Implementation details of CPM can be seen at [4].

D. Comparison method for environment recognition

We compared SLAMER's environment recognition accuracy with a simple map-based recognition method. The simple method uses SLFM. SLFM is calculated according to the estimated pose by SLAMER. The maximum likelihood class is assigned as the estimated class to each scan point.

E. Results

Table I shows the comparison results. The units of pose and angle estimation errors are meters and degrees. The performance of LFM and SLFM were not similar, but it was difficult to say which one is superior. This result indicated that the simple use of the environment recognition results cannot yield performance improvement. The performance could be improved if an accurate recognition method can be available.

SLAMER outperformed LFM and SLFM in all the sequences because SLAMER can utilize the environment recognition results even though SLFM cannot. This result showed that SLAMER can cope with uncertainty in the environment recognition.

However, SLAMER was not superior to CPM in all the sequences in terms of accuracy. This result is natural if we consider the background of their modeling. SLAMER has an advantage besides localization performance improvement. We discuss regarding them in Section V-F.

Table I also shows the environment recognition accuracy by the network (ER Acc), simple method (Map-based ER ACC), and SLAMER (SLAMER ER Acc), respectively.

TABLE I

POSE/ANGLE ESTIMATION ERRORS BY LFM, SLFM, CPM, AND SLAMER ON THE SEMANTICKITTI DATASET.

	Sequence	03	04	05	06	07	09	10
LFM	Ave	15.44 / 0.38	12.44 / 0.35	7.88 / 0.22	25.32 / 0.53	29.28 / 0.44	18.42 / 0.35	6.16 / 0.20
	Std	14.58 / 0.40	11.91 / 0.30	7.23 / 0.23	46.57 / 1.29	22.89 / 0.39	55.81 / 0.74	4.60 / 0.22
	Max	89.10 / 3.32	61.16 / 1.90	98.15 / 2.40	278.01 / 15.11	140.36 / 2.34	587.30 / 11.21	48.31 / 2.34
SLFM	Ave	11.47 / 0.32	14.75 / 0.26	9.28 / 0.26	17.88 / 0.42	11.23 / 0.33	14.06 / 0.39	9.81 / 0.31
	Std	7.79 / 0.29	12.24 / 0.24	6.79 / 0.25	13.56 / 0.40	7.52 / 0.29	10.86 / 0.35	6.99 / 0.28
	Max	54.18 / 1.88	77.01 / 1.88	65.46 / 2.44	100.39 / 2.69	64.94 / 2.20	104.53 / 3.32	86.76 / 2.29
CPM	Ave	7.80 / 0.20	7.52 / 0.14	5.42 / 0.14	9.58 / 0.22	6.75 / 0.19	7.75 / 0.19	5.65 / 0.17
	Std	5.10 / 0.18	5.57 / 0.12	3.32 / 0.15	6.71 / 0.20	5.37 / 0.19	5.09 / 0.17	3.85 / 0.16
	Max	38.36 / 1.61	39.79 / 0.78	25.45 / 1.29	58.98 / 1.39	69.14 / 1.62	46.84 / 1.31	41.25 / 1.27
SLAMER	Ave	9.97 / 0.25	8.69 / 0.15	7.60 / 0.20	13.41 / 0.28	10.51 / 0.28	12.12 / 0.31	7.85 / 0.24
	Std	7.34 / 0.24	6.29 / 0.13	5.13 / 0.18	8.76 / 0.25	6.67 / 0.24	9.49 / 0.27	6.06 / 0.23
	Max	68.82 / 2.54	43.52 / 0.70	37.21 / 1.37	65.33 / 1.69	45.42 / 1.72	131.92 / 2.50	43.82 / 2.05
ER Acc	Ave	76.90 %	77.35 %	79.04 %	74.32 %	85.30 %	78.61 %	75.21 %
	Std	6.50 %	5.65 %	4.07 %	6.76 %	4.67 %	5.30 %	5.80 %
	Min	60.33 %	60.17 %	63.49 %	54.71 %	70.51 %	57.36 %	44.80 %
	Max	92.09 %	91.01 %	89.84 %	86.16 %	95.07 %	90.41 %	85.99 %
Map-based ER Acc	Ave	83.52 %	87.34 %	78.25 %	75.59 %	79.75 %	77.59 %	76.15 %
	Std	6.21 %	6.34 %	7.94 %	10.36 %	8.61 %	7.52 %	8.61 %
	Min	70.20 %	74.21 %	46.90 %	43.25 %	44.95 %	52.29 %	40.34 %
	Max	95.69 %	96.77 %	94.48 %	92.84 %	93.35 %	95.27 %	94.76 %
SLAMER ER Acc	Ave	85.72 %	90.15 %	83.86 %	81.17 %	89.02 %	80.32 %	79.21 %
	Std	6.42 %	5.28 %	7.79 %	8.80 %	2.95 %	7.80 %	6.77 %
	Min	70.22 %	75.63 %	49.28 %	47.41 %	76.51 %	53.38 %	54.41 %
	Max	95.63 %	96.64 %	95.69 %	94.31 %	95.82 %	95.82 %	94.68 %

SLAMER achieved accurate recognition more than the network in all the sequences even when the simple map-based method sometimes degradated. However, the minimum recognition accuracy by SLAMER was sometimes bad more than that of the network. This was yielded by an inaccurate localization result, but SLAMER could improved the average recognition accuracy. In addition, we could confirm that the simple-based environment recognition is not effective to improve the recognition accuracy. From these results, we could reveal that SLAMER can cope with environment recognition uncertainty.

F. Discussion

As can be seen from Table I, CPM outperformed all the methods in terms of accuracy. We discuss why SLAMER cannot outperform CPM while respecting their advantages.

The model that derives CPM assumes that the environment recognition results, $\hat{\mathbf{c}}$, depend on the pose, sensor measurement, and map. This dependency enables to consider how the environmental object classes are predicted while considering relationship between the pose, sensor measurement, and map. This relationship is significant for localization. Hence, CPM, $p(\hat{\mathbf{c}}_t|\mathbf{x}_t, \mathbf{z}_t, \mathbf{m}, \Theta)$, can utilize the class prediction results to improve the localization accuracy.

However, the likelihood distribution used in SLAMER, $\sum_{\mathbf{c}_t} \{p(\hat{\mathbf{c}}|\mathbf{c}_t, \mathbf{z}_t, \Theta) p(\mathbf{c}_t|\mathbf{x}_t, \mathbf{m})\} p(\mathbf{z}_t|\mathbf{x}_t, \mathbf{m})$, does not have such relationship. The environment recognition model, $p(\hat{\mathbf{c}}|\mathbf{c}_t, \mathbf{z}_t, \Theta)$, only considers the sensor measurement and the true classes. Consequently, the likelihood distribution cannot have significant effect to improve the localization accuracy. However, the likelihood distribution includes the prior distribution over the true object classes. This contributes to improve the localization performance while utilizing the environment recognition results.

In addition, since the likelihood distribution does not have the relationship considered in CPM, objects which might not have better influence for localization can also be treated in the SLAMER framework. For example spatial objects such as no entry lines can be handled in the framework. This advantage is shown in the next section.

VI. INDOOR EXPERIMENTS

In this section, we show qualitative performance of 2D LiDAR-based SLAMER with our experimental platform. This 2D LiDAR-based implementation is publicly available[1].

A. Experimental equipment

The experimental platform is shown in Fig. 2. We used i-Cart mini robot equipped with URM-40LC/LCN-EW LiDAR. This robot is equipped with wheel encoders and we used it as INS. We used ROS gmapping to build a 2D map and manually created the semantic map. Figure 3 shows the example map. In the experiment, door (pink), glass door (cyan), and no entry line (red) objects are assigned. Based on the objects, we consider following environmental object classes; open door, close door, open glass door, close glass door, no entry line, free space, and others.

200 particles were used to implement RBPF. The hyper-parameters shown in Eq. (6) were set to $a_1 = 2$ and $a_2 = 2$.

B. Object recognition from 2D LiDAR measurement

In this experiment, we use a simple line object detection method. A 2D LiDAR scan image is first made and the probabilistic Hough transform implemented in OpenCV is applied to the image to detect line objects. Then, rate of the scan points included in each line is calculated. Probability

[1] https://github.com/NaokiAkai/als_ros

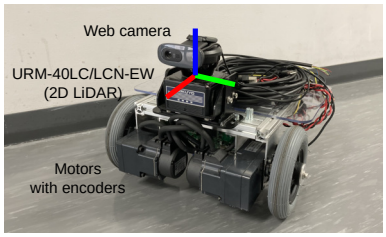

Fig. 2. The experimental platform (iCart-mini). The red, green, and blue lines indicate x, y, and z axes of the 2D LiDAR. The web camera is just used for visualization as shown in Fig. 1.

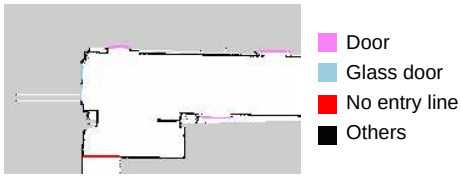

Fig. 3. An example of the semantic map. The map shown in Fig. 1 is enlarged at the bottom.

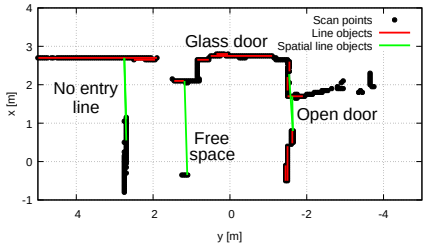

Fig. 4. An example of the line object detection result in the 2D LiDAR coordinates. This example is extracted from the data shown in Fig. 1.

over the environmental object classes is calculated based on the rate. Of course, this rate-based probability calculation is inaccurate and it does not have an important role in this experiment. Hence, we implemented a simple rule-based classification based on the rate.

We also consider spatial objects that do not have physical shape but has environmental meanings such as open doors and no entry lines. The spatial objects are also detected as lines and these lines are detected based on dominant orientation of the scan points. The spatial line objects also classified using a simple rule.

Figure 4 shows an example of the line object detection result. This example is extracted from the data shown in the bottom of Fig. 1. The black points depict the scan points and the red and green lines depict the physical and spatial line objects. Parallel lines to the scan points were detected as the spatial line objects.

C. Results and discussion

The bottom of Fig. 1 shows the qualitative results of SLAMER's estimate. As can be seen from the figure, we could confirm that SLAMER can recognize the environmental objects. However, as we mentioned in the previous subsection, the classification based on the points rate was

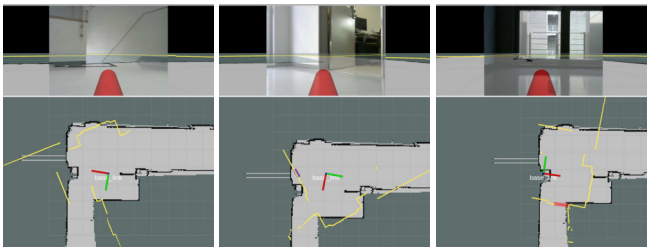

Fig. 5. Environment object recognition results in miss localization cases.

inaccurate. This means that the map-based recognition was dominant in the cases shown in Fig. 1 since localization has succeeded. We discuss SLAMER's performance based on inaccurate localization cases because performance regarding uncertainty consideration can be confirmed in such cases.

Figure 5 shows the environmental object recognition results in miss localization cases. The left and middle figures of Fig. 5 show better results of SLAMER's estimate. The object recognition method presented in Section VI-B recognized line objects as shown in Fig. 4; however, SLAMER did not output any environmental objects. In particular, the open door was not recognized as the no entry line even though its position on the map was closed to the no entry line. This result revealed that SLAMER can cope with uncertainties in localization, environment recognition, and mapping.

However, SLAMER does not work in some cases. The right figure of Fig. 5 shows an miss recognition result. The position of the open door was exactly overlapped with that of the no entry line and the open door was recognized as the no entry line (red line). Even though SLAMER can perform environmental object recognition with the Bayes filter, SLAMER cannot work in such a worst case. However, occuring such a worst overlap is seldom rare. In addition, SLAMER can have possibility to overcome such cases because it can use the environmental object recognition model to update the map-based prior.

VII. Conclusion

This paper presented SLAMER, the simultaneous localization and map-assisted object recognition method. SLAMER is the probabilistic model to cope with uncertainties included in localization and environmental object recognition. In this paper, we demonstrated 2D- and 3D-LiDAR-based implementation of SLAMER. For the 3D-LiDAR-based demonstration, we used the SemanticKITTI dataset and showed that SLAMER improved both localization and environmental object recognition accuracy. For the 2D-LiDAR-based demonstration, we used the indoor mobile robot and showed that SLAMER realized recognition of unmeasurable environmental objects such as open doors and no entry lines. We also showed that SLAMER cannot outperform the class prediction model (CPM) presented in [4] in terms of localization accuracy. However, SLAMER can have an advantage that objects which might not have significant influence to localization can be handled. Outperforming CPM while ensuring the advantage of SLAMER is our future work.

ACKNOWLEDGMENT

Support for this work was given by the Toyota Motor Corporation (TMC) and JSPS KAKENHI under Grant 18K13727. However, note that this article solely reflects the opinions and conclusions of its author and not TMC or any other Toyota entity.

REFERENCES

[1] N. Akai, K. Yamauchi, K. Inoue, Y. Kakigi, Y. Abe, and K. Ozaki. Development of mobile robot "SARA" that completed mission in real world robot challenge 2014. *Journal of Robotics and Mechatronics*, 27(4):327–336, 2015.

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

[3] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall. SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[4] N. Akai, T. Hirayama, and H. Murase. Semantic localization considering uncertainty of object recognition. *IEEE Robotics and Automation Letters*, 5(3):4384–4391, 2020.

[5] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12):2481–2495, 2017.

[6] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss. RangeNet++: Fast and accurate lidar semantic segmentation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4213–4220, 2019.

[7] E. Stenborg, C. Toft, and L. Hammarstrand. Long-term visual localization using semantically segmented images. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 6484–6490, 2018.

[8] J.L. Schönberger, M. Pollefeys, A. Geiger, and T. Sattler. Semantic visual localization. In *Proceedings of the IEEE / CVF Computer Vision and Pattern Recognition (CVPR)*, pages 6896–6906, 2018.

[9] S. Wen, P. Li, Y. Zhao, H. Zhang, F. Sun, and Z. Wang. Semantic visual SLAM in dynamic environment. *Autonomous Robots*, 45:493–504, 2021.

[10] L. Nicholson, M. Milford, and N. Sünderhauf. QuadricSLAM: Dual quadrics from object detections as landmarks in object-oriented SLAM. *IEEE Robotics and Automation Letters*, 4(1):1–8, 2019.

[11] S. Ren, K. He, R.B. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, pages 91–99, 2015.

[12] A. Zaganidis, M. Magnusson, T. Duckett, and G. Cielniak. Semantic-assisted 3D normal distributions transform for scan registration in environments with limited structure. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4064–4069, 2017.

[13] A. Zaganidis, L. Sun, T. Duckett, and G. Cielniak. Integrating deep semantic segmentation into 3-D point cloud registration. *IEEE Robotics and Automation Letters*, 3(4):2942–2949, 2018.

[14] X. Chen, A. Milioto, E. Palazzolo, P. Giguère, J. Behley, and C. Stachniss. SuMa++: Efficient LiDAR-based semantic SLAM. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4530–4537, 2019.

[15] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003.

[16] S.A. Parkison, L. Gan, M.G. Jadidi, and R.M. Eustice. Semantic iterative closest point through expectation-maximization. In *Proceedings of the British Machine Vision Conference*, 2018.

[17] J. Stechschulte and C. Heckman. Hidden Markov random field iterative closest point. *CoRR*, arXiv:1711.05864, 2017.

[18] F.T. Ramos, D. Fox, and H. Durrant-Whyte. CRF-Matching: Conditional random fields for feature-based scan matching. In *Robotics: Science and Systems (RSS)*. The MIT Press, 2007.

[19] S.L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas. Probabilistic data association for semantic SLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1722–1729, 2017.

[20] N. Atanasov, M. Zhu, K. Daniilidis, and G. J. Pappas. Localization from semantic observations via the matrix permanent. *The International Journal of Robotics Research*, 35(1–3):73–99, 2016.

[21] G.D. Tipaldi, D. Meyer-Delius, and W. Burgard. Lifelong localization in changing environments. *The International Journal of Robotics Research*, 32(14):1662–1678, 2013.

[22] M. Montemerlo, S. Thrun, D. Roller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1151–1156, 2003.

[23] C.-C. Wang, C. Thorpe, S. Thrun, M. Hebert, and H. Durrant-Whyte. Simultaneous localization, mapping and moving object tracking. *The International Journal of Robotics Research*, 26(9):889–916, 2007.

[24] C.M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag, 2006.

[25] N. Akai, T. Hirayama, and H. Murase. 3D Monte Carlo localization with efficient distance field representation for automated driving in dynamic environments. In *Proceedings oth the IEEE Intelligent Vehicles Symposium (IV)*, pages 1859–1866, 2020.