2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

Lifelong Localization in Semi-Dynamic Environment

Shifan Zhu[1], Xinyu Zhang[1,*], Shichun Guo[1], Jun Li[1], and Huaping Liu[2]

Abstract— **Mapping and localization in non-static environments are fundamental problems in robotics. Most of previous methods mainly focus on static and highly dynamic objects in the environment, which may suffer from localization failure in semi-dynamic scenarios without considering objects with lower dynamics, such as parked cars and stopped pedestrians. In this paper, we introduce semantic mapping and lifelong localization approaches to recognize semi-dynamic objects in non-static environments. We also propose a generic framework that can integrate mainstream object detection algorithms with mapping and localization algorithms. The mapping method combines an object detection algorithm and a SLAM algorithm to detect semi-dynamic objects and constructs a semantic map that only contains semi-dynamic objects in the environment. During navigation, the localization method can classify observation corresponding to static and non-static objects respectively and evaluate whether those semi-dynamic objects have moved, to reduce the weight of invalid observation and localization fluctuation. Real-world experiments show that the proposed method can improve the localization accuracy of mobile robots in non-static scenarios.**

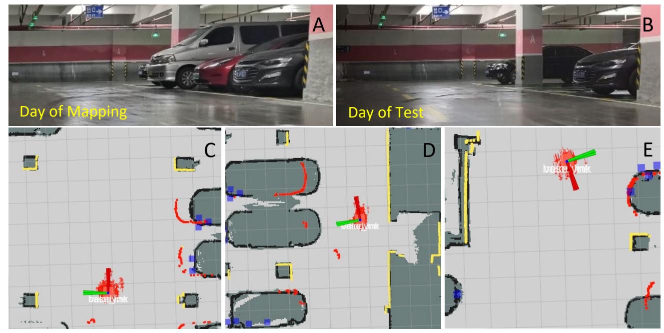

Fig. 1. Robot operates in the environment with both static objects (walls and pillars) and semi-dynamic objects (cars). The environment during mapping (A) and localization (B) is quite different due to the moving of cars. The figures (C, D, E) represent the observation during the localization process, the red LiDAR points correspond to semi-dynamic objects, which may or may not have been moved. The yellow LiDAR points correspond to static objects. The weight of current pose can be recalculated by the classification of observation.

I. INTRODUCTION

Simultaneous Localization and Mapping (SLAM) is a fundamental problem in mobile robotics[1], [2]. Over the past several decades, efforts have been made to extend its application from laboratory scenes to outdoor scenes, where the major difference is the dynamic degree[3], [4], [5].

Besides static objects, more frequently seen dynamic and semi-dynamic objects in outdoor environments may cause fluctuations to localization system, which pose crucial challenges to mapping and localization system [6], [7], [8], [9]. To solve this complex yet promising problem, researchers began to break the static world assumption, and the inevitable obstacle is to identify highly dynamic and semi-dynamic objects in the environment. By modeling and filtering out dynamic objects, precise mapping and localization results have been achieved in highly dynamic environments[10], [11]. However, in environments containing many semi-dynamic objects, mapping and localization are still open problems and remain unsolved due to the similarity of dynamic characteristics of static and semi-dynamic objects.

To tackle this problem, features that have less variation over time are selected. Methods that utilize semantic information for localization have been widely applied and proven effective [12], [13], [14]. Therefore, in this paper, we propose a related method to recognize and record the static and semi-dynamic objects in the environment, which can be divided into two parts: mapping static objects and semi-dynamic objects respectively, and robot localization using both static map and semi-dynamic map.

The mapping algorithm in this work extends the occupancy grid technique introduced by Hess et al. [15] to record semi-dynamic parts of the environment. It is capable of detecting both static and semi-dynamic objects in the environment and generating two maps: the static map and the semi-dynamic map. After filtering out the dynamic objects during the mapping process, the static map contains both static and semi-dynamic objects in the environment, while the semi-dynamic map only contains semi-dynamic objects.

The localization algorithm in this work optimizes the particle filter algorithm by using two maps simultaneously. It allows for the observation classification (static or semi-dynamic objects), as well as a determination of the distance the semi-dynamic objects have moved. Followed by adjustment of the weights of different objects, a more precise localization result can be achieved in semi-dynamic environments.

In summary, our main contributions are:

- We propose a semantic mapping method that can identify and record environment's semi-dynamic features.

This work was supported by the National High Technology Research and Development Program of China under Grant No. 2018YFE0204300, and the Beijing Science and Technology Plan Project (Z191100007419008), and the Guoqiang Research Institute Project (2019GQG1010), and the National Natural Science Foundation of China under Grant No. U1964203.

[1]S. Zhu, X. Zhang, S. Guo, J. Li are with the State Key Laboratory of Automotive Safety and Energy, and the School of Vehicle and Mobility, Tsinghua University, Beijing, 100084 China. (e-mail: shifzhu@gmail.com; xyzhang@tsinghua.edu.cn; shichunguo@gmail.com; lijun19580326@126.com).

[2]H. Liu is with the Department of Computer Science and Technology, Tsinghua University, Beijing, 100084 China. (e-mail: hpliu@mail.tsinghua.edu.cn).

*Author to whom correspondence should be addressed. (e-mail: xyzhang@tsinghua.edu.cn).

- We construct a robust localization system that can determine object categories according to the observation.
- We introduce a generic framework that can combine mainstream object detection and SLAM algorithms to achieve high localization accuracy regardless of position changes of semi-dynamic objects.

The rest of this article is organized as follows: Section II summarizes related works. Section III introduces the framework, mapping, and localization methods. Section IV evaluates the localization accuracy of the proposed methods based on a real-world environment. Section V summarizes our research and prospects for future work.

II. RELATED WORK

Mapping and localization in non-static environments is a challenging task that has received extensive attention recently. Several solutions have been attempted by modeling the static parts of the environment and filtering out dynamic objects[6]. Burgard et al. [16] proposed a distance filter to discard observation corresponding to dynamic objects. Hähnel et al. [17] presented a feature-based approach to track and map the people using joint probabilistic data association particle filters. Wang et al. [18] developed feature-based methods to filter out dynamic objects in the process of scan registration. Hähnel et al. [7] used an Expectation Maximization (EM) algorithm to identify static and dynamic parts of the environment off-line. These methods based on identifying and filtering out dynamic objects have proved robust in highly dynamic environments. However, they discard valuable observation data and cannot deal with semi-dynamic objects that move less frequently.

To solve this problem, approaches that maintained more than one map to represent the environment have been proposed. Wolf et al. [19] maintained two maps that represent dynamic and static objects separately. Krajník et al. [20] built new maps on the fly with a separate SLAM process, which ensures that the new map is consistent with the previous one. The newly-built maps are integrated into a spatiotemporal model that represents environment dynamics. Gallagher et al. [21] built separate maps for each object. Dynamic objects detected via laser observations are saved in lists with positions and orientations. Although these approaches utilize dynamic information and construct a map that better represents the current environment, they still share the same limitation of static assumption as previous methods.

To overcome the limitations of the static world assumptions, approaches aimed to find a model to represent the dynamics have been reported. Biber et al. [22] proposed a model that represents the environment over multiple timescales simultaneously. Montesano et al. [23] distinguished and utilized both static and dynamic information selectively to predict collision and improve navigation performance. Meyer-Delius et al. [24] developed a Hidden Markov Model to represent the dynamics of each grid and accurately model the occupancy rate over time. Akai et al. [25] designed an observation model where objects deviating from a certain threshold distance are considered as dynamic obstacles. However, if the dynamic objects are relatively close to the position of static objects during the map process, this assumption may fail. Li et al. [26] introduced a particle filter method to account for moving dynamic obstacles and used an active localization system to reduce dynamics. Ding et al. [27] designed a change detection module to display how likely change exists in each cell. However, these approaches require the robot to run multiple times on the same route to collect data from the dynamics of the environment.

In addition to the study of highly dynamic environments, low dynamic environments have also been specifically studied. Egger et al. [28] proposed a method for localization in long-term scale by using a sliding window to match current and old distinctive features and minimize the distances between them. Meyer-Delius et al. [29] divided objects in the environment into static objects, semi-static objects, dynamic objects. When the observations were not consistent with the static map, they used semi-static objects to build local maps that temporarily extend the reference map. Their later work [30] demonstrated their approach in semi-static environments by using Rao-Blackwellized particle filter with a hidden Markov model to do lifelong localization in changing environments. All the methods above focus on solving problems in situations where the static map cannot provide enough information, while our method commits to the situation where the static map provides wrong information and it also considers the dynamics of features.

Compared with traditional features, semantic features are invariant against environmental changes. Xiao et al. [31] combined semantic information with a feature-based dynamic SLAM system. They proposed a selection tracking algorithm to eliminate the dynamic objects and improve the robustness and accuracy. Bescos et al. [32] exploited RGB-D cameras and Mask R-CNN to remove the observation of dynamic objects in RGB-D cameras, which outperforms the accuracy of standard visual SLAM baselines in highly dynamic scenarios. Brasch et al. [33] used the semantic information in the pose estimation pipeline. Vineet et al. [34] introduced scene understanding to large-scale semantic scene reconstruction. They also presented a semantic fusion approach that can handle dynamic objects more effectively. Yu et al. [35] combined semantic segmentation network with moving consistency to improve the localization accuracy. Wang et al. [36] projected LiDAR data into images and applied 2D segmentation information of RGB images to 3-D LiDAR points. Stenborg et al. [12] used semantic segmentation to replace feature descriptors. Chen et al. [14] proposed a pose optimization method based on semantic features that simultaneously adjusted the tree position while estimating the robot pose. Previous semantic information based methods have been successfully applied in dynamic scenes, but neglecting the semi-dynamic characteristics represents the major limitation. In this paper, we propose a system that contains a mapping approach that utilizes an object detection algorithm to generate a semantic map. The following localization algorithm can be applied in semi-dynamic environments.

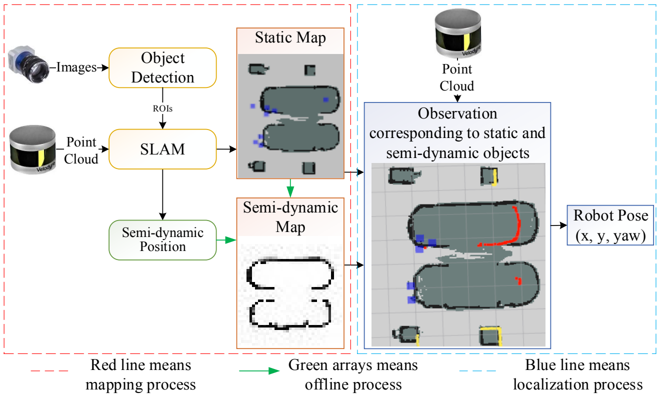

Fig. 2. The architecture of the proposed mapping and localization system. The static map contains two cars and four pillars, and the semi-dynamic map contains two cars. The blue dots in maps represent the semi-dynamic positions. The red LiDAR points and yellow LiDAR points correspond to semi-dynamic objects and static objects respectively.

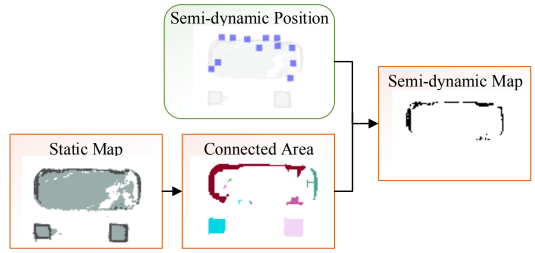

Fig. 3. A schematic diagram of the proposed mapping system. Semi-dynamic position (blue dots) and connected area (colored static map) produce the semi-dynamic map.

III. METHODOLOGY

This section describes the architecture of the proposed method. According to the classification from Meyer-Delius et al. [29], in this paper, the classifications are slightly revised as following:

- Static objects: objects that never change their position, like walls, shelves.
- Semi-dynamic objects: objects that are static during the mapping process but may be moved during the localization process, like chairs, parked cars.
- Dynamic objects: objects that change their position frequently, like moving people, moving cars.

Figure 2 gives an overview of the mapping and localization framework. During the mapping process(red dashed line), an object detection algorithm (Yolov3 [37]) is selected to recognize the semi-dynamic objects. In the SLAM thread, we exploit a SLAM algorithm (Cartographer [15]) to generate a static map, followed by projecting LiDAR points into the image. The given semi-dynamic position in global coordinate can later facilitate creating a semi-dynamic map m_d from the static map m_s. The static map m_s contains information of both static and semi-dynamic objects, while the semi-dynamic map m_d only contains information of semi-dynamic objects. Moreover, the semi-dynamic map is generated offline without increasing the complexity of the mapping process. During the localization process (blue dashed line), m_s and m_d are simultaneously loaded to the localization system. By comparing the distance of observation, we can obtain observation corresponding to static objects (yellow points) and semi-dynamic objects (red points). Information from both m_s and m_d is incorporated to localize the robot's pose, depending on whether and how far the object is moved. Precise localization can be achieved by reducing the weight of observation corresponding to semi-dynamic objects.

A. Mapping Approach

1) Static Map Update: The recognition of dynamic objects is achieved based on the probability that objects appear at the same location. Specifically, a certain area is dynamic if its status changes between occupied and free. In this paper, a probabilistic method was utilized to define the problem as:

$$P(s_t|z_{1:t}, s_{t-1}) \tag{1}$$

where $z_{1:t}$ is the sensor data from time 1 to time t, s_{t-1} is previous static map, s_t is current static map. By applying Bayes rule to Eq. 1 and eliminating some unnecessary terms we can get an inverse observation model that can only record the static parts of environments. Specifically, the grid will be recorded if the status of a grid is changing from unknown to occupied or is staying occupied. Please refer to [15][19] for more detailed information about inverse observation model.

2) Semi-dynamic Map Update: Observation z_t is classified into observation of static objects z_t^s or observation of semi-dynamic objects z_t^d. For this step, Yolov3 was applied to get the classification information. By calibrating the extrinsic parameters between LiDAR and camera, LiDAR points can be projected into these bounding boxes. However, it cannot provide accurate dimensional information because the LiDAR points at the edge of the bounding box may correspond to other objects. To solve this problem, only the points in the center of the bounding box are adopted to represent the position information of semi-dynamic objects, which excludes dimension information. With the position information, the following the calculation of connected areas on the static map gives a more accurate semi-dynamic map.

The blue dots in Figure 3 indicate the semi-dynamic positions. One or several connected areas indicate a single object. To supplement the dimension of semi-dynamic objects, we use a relatively low threshold to obtain the binary static map, followed by calculation of connected area. The integration of semi-dynamic positions and connected areas produces the semi-dynamic map. Only the semi-dynamic positions are obtained when constructing the static map, other processes in Figure 3 are offline which will not increase the computation complexity in the mapping process. For more experimental results, please refer to Part IV.

B. Localization Approach

For the localization process, the Monte Carlo method is used to validate the proposed idea by using odometry u_t and observation z_t recursively. We improve the localization accuracy by simultaneously loading m_s and m_d, which can

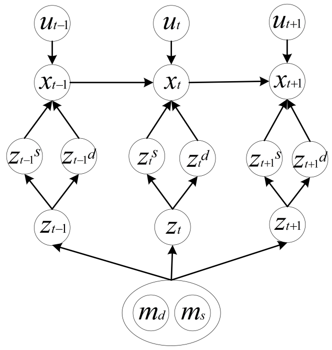

Fig. 4. Graphical model of the localization process.

identify and reduce the weight of observation corresponding to moved semi-dynamic objects. Figure 4 illustrates the graphical model of proposed localization approach, which has sensor observations z_t, static map m_s, semi-dynamic map m_d, odometry u_t, pose at time $t-1$ as input. The localization problem can be denoted as:

$$P(x_t|z_t, u_t, x_{t-1}, m_s, m_d) \tag{2}$$

By applying Bayes rule to Eq. 2, it can be expressed as:

$$\begin{aligned} &P(x_t|z_t, u_t, x_{t-1}, m_s, m_d) \\ &= \eta P(z_t|x_t, m_s, m_d) P(x_t|u_t, x_{t-1}) \end{aligned} \tag{3}$$

where η is a normalization constant. To further reduce the weight of moved objects, the classification of observation should be calculated based on m_s and m_d. The Eq. 3 can further be defined as:

$$\begin{aligned} &P(x_t|z_t, u_t, x_{t-1}, m_s, m_d) \\ &= \eta P(z_t^s, z_t^d|x_t, m_s, m_d) P(x_t|u_t, x_{t-1}) \end{aligned} \tag{4}$$

The major improvement of our method is that two maps are used simultaneously to classify the observation corresponding to semi-dynamic objects and static objects. The classification can be calculated by comparing the nearest distance between each LiDAR point and two maps. Let d_i^s represent the distance from point i to the nearest grid in the static map, let d_i^d represent the distance from point i to the nearest grid in the semi-dynamic map. If the difference between d_i^d and d_i^s is smaller than a threshold, current LiDAR points correspond to a semi-dynamic object, otherwise, the LiDAR point corresponds to a static object. The given result during the localization process (Figure 2) reflects semi-dynamic objects (red LiDAR points) and static objects (yellow LiDAR points). After classification, the weight of observation corresponding to semi-dynamic objects is reduced according to the distance they have moved. The decreasing ratio can be defined as $f(z_t, m_s, m_d, d_i^s, d_i^d)$. The result of the function is between zero and one, indicating how much the weight of current observation should be reduced. In this work, the positions of these semi-dynamic objects are supposed to follow a gaussian distribution. Let F represent the decreasing ratio $f(z_t, m_s, m_d, d_i^s, d_i^d)$, which can be

Algorithm 1 Weight of pose

Require: Observation z_t, Static map m_s, Semi-dynamic map m_d;
Ensure: Weight of current pose;

$sig = 2 * sigma * sigma$
$weight = 0$
for each $lidarpoint$ **do**
    if $(z_i \neq z_{max})$ **then**
        $x = pose\_x + range * cos\theta$
        $y = pose\_y + range * sin\theta$
        $dist_s = distance\ to\ nearest\ obstacles\ in\ m_s.$
        $dist_d = distance\ to\ nearest\ obstacles\ in\ m_d.$
        if $abs(dist_i^s - dist_i^d) < \epsilon_1$ and $dist_i^d > \epsilon_2$ **then**
            $coef = exp(-(dist_i^d)^2/sig)$
        else
            $coef = 1$
        end if
    end if
    $weight \mathrel{+}= coef * p\,(z_t \mid x_t, m_s, m_d)$
end for
return $weight$

defined as:

$$F = \begin{cases} \exp\left(-\frac{(d_i^d)^2}{2\sigma^2}\right) & \text{if } \left|d_i^d - d_i^s\right| < \epsilon_1 \text{ and } d_i^d > \epsilon_2 \\ 1 & \text{otherwise} \end{cases} \tag{5}$$

where ϵ_1 is a relatively small distance, taking 0.1 in this paper. ϵ_2 is the threshold at which an object is considered to be semi-dynamic, taking 0.3 in this paper. Therefore, the final weight of current position can be computed as:

$$\omega = f\left(z_t, m_s, m_d, d_i^s, d_i^d\right) * p\left(z_t \mid x_t, m_s, m_d\right) \tag{6}$$

where $p(z_t|x_t, m_s, m_d)$ is the observation model. In the paper, the likelihood observation model is chosen to calculate the weight of current position. The pose estimation can be more accurate by using both the static map and semi-dynamic map. The algorithm of calculating the weight of current pose is summarized in Algorithm 1.

One of the major advantages is that the three components in this framework, object detection, mapping, and localization algorithm can be replaced by other corresponding mainstream algorithms.

IV. EXPERIMENTS

In order to validate the approach proposed in this paper, experimental tests have been done using a real robot. The robot (Figure 6) utilized in this work is equipped with a 16-beam Velodyne LiDAR (Only one beam is used during mapping and localization), a camera, an inertial measurement unit, and encoders.

The parking lot was selected as the experimental scenario because the parking lot changed significantly every individual run. It consists of four rooms and the left room in the map frame usually shows more cars compared with others. The

Fig. 5. Top panel (A, B, C) show the static maps constructed using data collected in every individual run. (a', b', c') are the zoomed-in views of the area (a, b, c) where semi-dynamic objects (cars) change frequently. Bottom panel (D, E, F) show the semi-dynamic maps constructed using information from static maps and semi-dynamic positions.

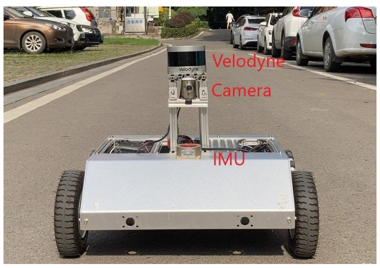

Fig. 6. The experimental platform.

robot started in the left room and went around the parking lot and back to its original place. Due to the particularity of the experimental scenarios, it is hard to directly get ground truth poses. We recorded the robot's poses during the mapping process as ground truth, and these poses during the localization process are also recorded using the map built on other days. The validation of proposed approach has been proved by comparing the localization accuracy without and with the proposed approach. The localization accuracy of using the proposed method in an altered environment was also compared to that of not using the proposed method in an unaltered environment.

A. Mapping in Parking Lot Scene

For the mapping process, we steered the robot to collect data every few days. The data collected during the individual run were used to build m_s. m_d is constructed offline by using m_s and positions of semi-dynamic objects.

The final constructed m_s and m_d are shown in Figure 5. In these static maps, some areas are distinct owing to the moving of cars, while the other parts are static all the time. m_d contains most of the semi-dynamic objects in the environment, and the missing semi-dynamic objects correspond to areas that are rarely observed, which have less impact on the localization accuracy. By applying the inverse observation model, the highly dynamic objects such as moving pedestrians and cars are automatically removed during mapping process.

TABLE I

QUANTITATIVE ANALYSIS OF POSE ERROR.

	MCL+−	MCL++	MCL−−	MCL−+	MCL*
Max [m]	0.516	0.336	0.336	0.327	0.423
Mean [m]	0.129	0.083	0.064	0.065	0.087
Min [m]	0.007	0.001	0.002	0.001	0.002
RMSE [m]	0.154	0.100	0.080	0.081	0.104
Std [m]	0.084	0.056	0.047	0.049	0.056

B. Localization in Non-Static Environment

The localization accuracy with and without our algorithm in both altered and unaltered semi-dynamic environments is compared (Figure 7ABCD). In addition, we also present the localization accuracy of original Monte Carlo localization without our method utilizing a pure static map (Figure 7E). All semi-dynamic objects, such as parked cars, were artificially removed. The poses during the mapping process were selected as ground truth, and the poses during the localization process were also recorded using maps built on other days. The algorithm Evo [38] was used to evaluate the accuracy of our approach offline. Figure 7 shows the qualitative result of localization accuracy and Table I lists the quantitative comparison of the pose error. The MCL+− and MCL++ mean in an altered environment using Monte Carlo algorithm without and with our method (The first plus or minus subscript indicates whether the environment of the localization process has changed or not, and the second plus or minus subscript indicates whether the localization process uses our method or not). Therefore, the MCL−− and MCL−+ mean in an unaltered environment using Monte Carlo algorithm without and with our method. MCL* means using Monte Carlo algorithm without our method, using a pure static map. The pure static map is shown in Figure 8 where the semi-dynamic objects are artificially removed from the map. Further, Figure 7ABCDE correspond to MCL+−, MCL++, MCL−−, MCL−+, MCL* in Table I respectively.

In Figure 7A and 7B, our algorithm significantly decreases the deviation in altered environments, giving a lower mean error (0.083m versus 0.129m) and max error (0.336m versus

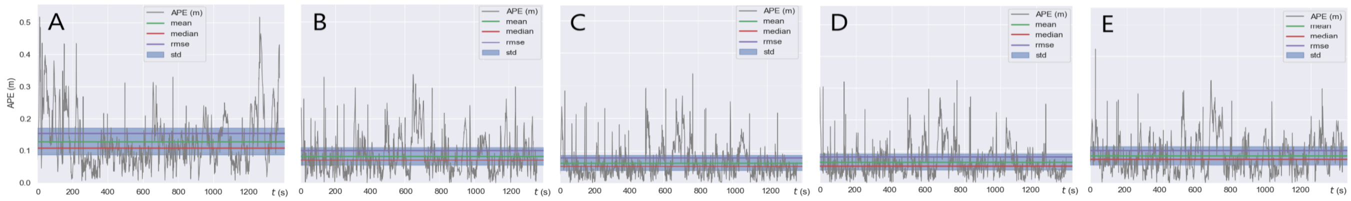

Fig. 7. Qualitative analysis of pose error. A and B show the localization accuracy without and with proposed method in an altered semi-dynamic environment. C and D show the localization accuracy without and with proposed method in an unaltered environment. E shows the localization accuracy of original Monte Carlo without proposed method using a pure static map.

Fig. 8. The pure static map. Semi-dynamic objects are artificially removed.

0.516m). According to the maps shown in Figure 5 and the route of the robot, it encounters more semi-dynamic objects during the first 300 seconds and the last 200 seconds and the localization accuracy fluctuates significantly in these places (Figure 7A), which proves that the original Monte Carlo algorithm has low performance in altered environments. On the contrary, the results are more accurate after applying our algorithm (Figure 7B).

We also developed experiments to evaluate the algorithm performance in static environments. In Figure 7C and 7D, our algorithm can also be applied in unaltered environments, giving relatively the same accuracy (0.327m versus 0.336, 0.065m versus 0.064) compared with traditional Monte Carlo localization method. That proves the versatility of the proposed methods in the environment with both static and semi-dynamic objects.

Besides, we also carried out experiments to test the localization performance of the original Monte Carlo localization algorithm in an altered environment, but the semi-dynamic objects are artificially removed from the map. Thus, the map only constains static objects of the environment (Figure 8). The robot utilizes the static objects in the environment to localize the pose. The localization accuracy can be seen in Figure 7E. It is reasonable that it has a lower accuracy as all the semi-dynamic information is discarded.

In addition, m_d is obtained offline. Therefore, during the localization process the robot only needs to compare the distance between two maps, so the computational complexity is negligible. It should be noted that the absolute accuracy of Monte Carlo algorithm is not the focus of this paper. We aim to introduce our method into mainstream mapping and localization algorithms to obtain a more accurate result without sacrificing efficiency. And the experiment results have proven the better robustness, higher accuracy, and versatility of our approach in both static and semi-dynamic environments.

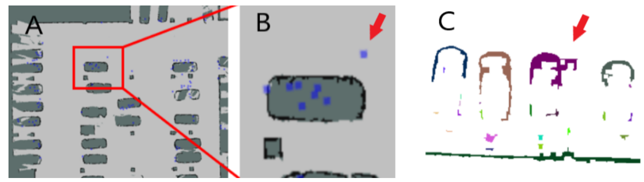

Fig. 9. Incorrect semi-dynamic position detection (Figure A and Figure B). B is the zoomed-in view of the red box area in A. The blue dot pointed by the red arrow in B is incorrectly regarded as a semi-dynamic position; Incorrect connected areas (Figure C). The place pointed by the red arrow in C is a pillar, which should be considered as a static object.

C. Some Tricky Situations

During the process of getting semi-dynamic positions, there may be some semi-dynamic positions that hit the wrong place (Figure 9A and B). However, the incorrect position does not have influence on m_d generation as it has no corresponding connected area.

During the process of calculating the connected area, some static areas will be mistakenly regarded as semi-dynamic areas. In Figure 9C, the pillar is mistaken for a semi-dynamic object, nevertheless, the static characteristic of these objects (pillar and wall) makes the weight of this area a constant. Therefore, the wrong connected area does not jeopardize the localization accuracy.

V. CONCLUSION AND FUTURE WORK

This paper introduced a novel semantic map generation method and a localization method that can deal with the semi-dynamic environment. By fusing the camera and LiDAR, the presented method can autocratically detect and mark semi-dynamic objects during mapping. With this critical information the 2D semi-dynamic map is constructed for later navigation, and a posterior distribution-based pose estimation is designed. We also apply several evaluations in an underground parking garage with a mobile platform that equipped a camera, LiDAR, and IMU. Results show that our method is able to work in most cases. Based on this work, our framework can be easily extended into 3D modes.

In a further extension, we will investigate the use of other types of maps and different localization algorithms, such as point cloud map and NDT localization algorithm.

ACKNOWLEDGMENT

It is sponsored by Tsinghua University-Didi Joint Research Center for Future Mobility.

REFERENCES

[1] S. Thrun, W. Burgard, and D. Fox, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Autonomous Robots*, vol. 5, no. 3-4, pp. 253–271, 1998.

[2] J. D. Tardós, J. Neira, P. M. Newman, and J. J. Leonard, "Robust mapping and localization in indoor environments using sonar data," *The International Journal of Robotics Research*, vol. 21, no. 4, pp. 311–330, 2002.

[3] O. Aycard, P. Laroche, and F. Charpillet, "Mobile robot localization in dynamic environments using places recognition," in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146)*, vol. 4. IEEE, 1998, pp. 3135–3140.

[4] J. Andrade-Cetto and A. Sanfeliu, "Concurrent map building and localization on indoor dynamic environments," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, no. 03, pp. 361–374, 2002.

[5] G. Wan, X. Yang, R. Cai, H. Li, Y. Zhou, H. Wang, and S. Song, "Robust and precise vehicle localization based on multi-sensor fusion in diverse city scenes," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 4670–4677.

[6] D. Fox, W. Burgard, and S. Thrun, "Markov localization for mobile robots in dynamic environments," *Journal of artificial intelligence research*, vol. 11, pp. 391–427, 1999.

[7] D. Hahnel, R. Triebel, W. Burgard, and S. Thrun, "Map building with mobile robots in dynamic environments," in *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, vol. 2. IEEE, 2003, pp. 1557–1563.

[8] C. Linegar, W. Churchill, and P. Newman, "Work smart, not hard: Recalling relevant experiences for vast-scale but time-constrained localisation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 90–97.

[9] M. Aldibaja, N. Suganuma, and K. Yoneda, "Robust intensity-based localization method for autonomous driving on snow–wet road surface," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 5, pp. 2369–2378, 2017.

[10] R. Valencia, J. Saarinen, H. Andreasson, J. Vallvé, J. Andrade-Cetto, and A. J. Lilienthal, "Localization in highly dynamic environments using dual-timescale ndt-mcl," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 3956–3962.

[11] 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*, vol. 26, no. 9, pp. 889–916, 2007.

[12] E. Stenborg, C. Toft, and L. Hammarstrand, "Long-term visual localization using semantically segmented images," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 6484–6490.

[13] K. Doherty, D. Fourie, and J. Leonard, "Multimodal semantic slam with probabilistic data association," in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 2419–2425.

[14] S. W. Chen, G. V. Nardari, E. S. Lee, C. Qu, X. Liu, R. A. F. Romero, and V. Kumar, "Sloam: Semantic lidar odometry and mapping for forest inventory," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 612–619, 2020.

[15] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2d lidar slam," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1271–1278.

[16] W. Burgard, A. B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun, "Experiences with an interactive museum tour-guide robot," *Artificial intelligence*, vol. 114, no. 1-2, pp. 3–55, 1999.

[17] D. Hähnel, D. Schulz, and W. Burgard, "Map building with mobile robots in populated environments." in *IROS*, 2002, pp. 496–501.

[18] C.-C. Wang and C. Thorpe, "Simultaneous localization and mapping with detection and tracking of moving objects," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, vol. 3. IEEE, 2002, pp. 2918–2924.

[19] D. F. Wolf and G. S. Sukhatme, "Mobile robot simultaneous localization and mapping in dynamic environments," *Autonomous Robots*, vol. 19, no. 1, pp. 53–65, 2005.

[20] T. Krajník, J. P. Fentanes, M. Hanheide, and T. Duckett, "Persistent localization and life-long mapping in changing environments using the frequency map enhancement," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4558–4563.

[21] G. Gallagher, S. S. Srinivasa, J. A. Bagnell, and D. Ferguson, "Gatmo: A generalized approach to tracking movable objects," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 2043–2048.

[22] P. Biber, T. Duckett *et al.*, "Dynamic maps for long-term operation of mobile service robots," in *Robotics: science and systems*, 2005, pp. 17–24.

[23] L. Montesano, J. Minguez, and L. Montano, "Modeling the static and the dynamic parts of the environment to improve sensor-based navigation," in *Proceedings of the 2005 IEEE international conference on robotics and automation*. IEEE, 2005, pp. 4556–4562.

[24] D. Meyer-Delius, M. Beinhofer, and W. Burgard, "Occupancy grid models for robot mapping in changing environments," in *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[25] N. Akai, L. Y. Morales, and H. Murase, "Mobile robot localization considering class of sensor observations," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 3159–3166.

[26] A. Q. Li, M. Xanthidis, J. M. O'Kane, and I. Rekleitis, "Active localization with dynamic obstacles," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 1902–1909.

[27] W. Ding, S. Hou, H. Gao, G. Wan, and S. Song, "Lidar inertial odometry aided robust lidar localization system in changing city scenes," in *2020 Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020.

[28] P. Egger, P. V. Borges, G. Catt, A. Pfrunder, R. Siegwart, and R. Dubé, "Posemap: Lifelong, multi-environment 3d lidar localization," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 3430–3437.

[29] D. Meyer-Delius, J. Hess, G. Grisetti, and W. Burgard, "Temporary maps for robust localization in semi-static environments," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 5750–5755.

[30] G. D. Tipaldi, D. Meyer-Delius, and W. Burgard, "Lifelong localization in changing environments," *The International Journal of Robotics Research*, vol. 32, no. 14, pp. 1662–1678, 2013.

[31] L. Xiao, J. Wang, X. Qiu, Z. Rong, and X. Zou, "Dynamic-slam: Semantic monocular visual localization and mapping based on deep learning in dynamic environment," *Robotics and Autonomous Systems*, vol. 117, pp. 1–16, 2019.

[32] B. Bescos, J. M. Fácil, J. Civera, and J. Neira, "Dynaslam: Tracking, mapping, and inpainting in dynamic scenes," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 4076–4083, 2018.

[33] N. Brasch, A. Bozic, J. Lallemand, and F. Tombari, "Semantic monocular slam for highly dynamic environments," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 393–400.

[34] V. Vineet, O. Miksik, M. Lidegaard, M. Nießner, S. Golodetz, V. A. Prisacariu, O. Kähler, D. W. Murray, S. Izadi, P. Pérez *et al.*, "Incremental dense semantic stereo fusion for large-scale semantic scene reconstruction," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 75–82.

[35] C. Yu, Z. Liu, X.-J. Liu, F. Xie, Y. Yang, Q. Wei, and Q. Fei, "Ds-slam: A semantic visual slam towards dynamic environments," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1168–1174.

[36] B. H. Wang, W.-L. Chao, Y. Wang, B. Hariharan, K. Q. Weinberger, and M. Campbell, "Ldls: 3-d object segmentation through label diffusion from 2-d images," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2902–2909, 2019.

[37] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[38] M. Grupp, "evo: Python package for the evaluation of odometry and slam," *url: https://github. com/MichaelGrupp/evo*, 2017.