IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 17, 2019, accepted January 4, 2020, date of publication January 9, 2020, date of current version January 17, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2965300

Hybrid Maps Enhanced Localization System for Mobile Manipulator in Harsh Manufacturing Workshop

GEN LI, YU HUANG, XIAOLONG ZHANG, CHAO LIU, WENJUN SHAO, LIQUAN JIANG, AND JIE MENG
School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China

Corresponding author: Jie Meng (mengjie_10@hust.edu.cn)

This work was supported in part by the Guangdong Major Science and Technology Project under Grant 2019B090919003, in part by the National Natural Science Foundation of China under Grant 51805190, in part by the Science and Technology Planning Project of Guangdong Province under Grant 2017B090913001, in part by the Dongguan Innovative Research Team Program under Grant 201536000100031, and in part by the Launch fund of the School of Mechanical Science and Engineering, Huazhong University of Science and Technology, under Grant 100-0199100041.

ABSTRACT With excellent mobility and flexibility, mobile manipulators have great potential for loading and unloading tasks of numerical control machine tools (CNC) in manufacturing workshops. However, because of the rough and oily ground, dynamic obstacles and the convex plate of a CNC, harsh manufacturing workshop poses a huge challenge to the localization system of an autonomous mobile manipulator. To address the above problem, this paper presents a hybrid maps enhanced localization system which mainly consists of a global localization method and a pose tracking method. Hybrid maps including hybrid grid map, multi-resolution likelihood fields (MLFs) and hybrid point map are constructed to efficaciously model the harsh environment and to improve localization performance. Our global localization method employs the convex hull sampling to spares dense Lidar data and the MLFs based branch and bound (BnB) search to speed up global search. To achieve real-time localization reliably and accurately, our pose tracking method seamlessly combines the BnB search and the adaptive Monte Carlo localization, and the Iterative Closest Point (ICP) based scan matching using the hybrid point map is adopted for higher accuracy. In addition, a distance filter improved by unscented transform is integrated into the pose tracking process to mitigate the influence of dynamic obstacles. The developed localization system is evaluated through different experiments including two weeks of loading and unloading tasks in a real manufacturing scenario, resulting in superior localization performance.

INDEX TERMS Mobile manipulator, localization system, hybrid maps, global localization, pose tracking, manufacturing workshop.

I. INTRODUCTION

Loading and unloading workpieces for CNC is an essential process for a manufacturing workshop. To automate this process, mobile manipulator is considered as core equipment due to its flexibility and mobility [1], [2]. To this end, we have developed a four-wheel-steering and four-wheel-driving (4WS4WD) mobile manipulator shown in Fig. 1. It mainly consists of two components: a 4WS4WD mobile platform equipped with two Hokuyo UTM-30LX 2D Lidars and eight encoders for omnidirectional locomotion and a robot arm equipped with a camera for precisely loading and unloading workpieces. Once a mobile manipulator gets a service request from a CNC, it will autonomously navigate to a pre-taught position near that CNC and then complete its task with its robot arm.

Through our test, we found that only if the positioning error of the robot arm is less than 0.2mm, workpieces can be successfully placed into fixture. Therefore, we employ a QR code based visual positioning module to obtain the relative pose between the robot arm and a fixture with sub-millimeter accuracy. However, to ensure that a QR code

The associate editor coordinating the review of this manuscript and approving it for publication was Okyay Kaynak.

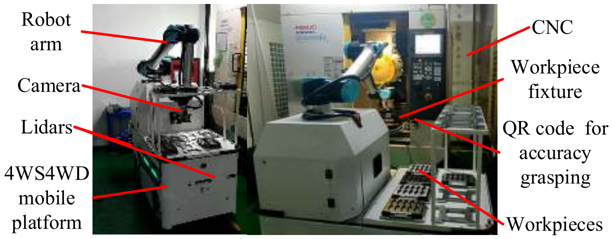


FIGURE 1. Self-developed mobile manipulator and its working scenario.

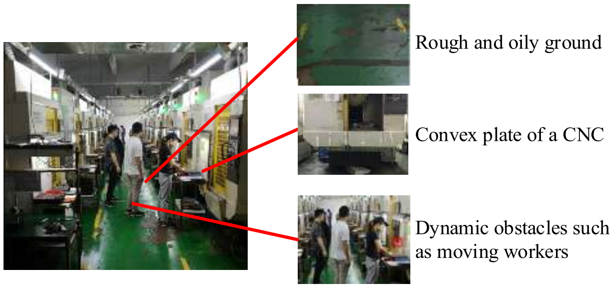


FIGURE 2. A typical manufacturing workshop: the right part shows three issues which challenge to mobile platform localization.

image is availably captured, the mobile platform must park within 20 mm of the pre-taught position each time. Therefore, it is crucial to achieve an accurate and reliable localization for the mobile platform. Some works have been done for robot localization in industry environments [3]–[5]. In their implementation, 2D Lidar is the key sensor due to its high frequency, high accuracy, low cost and small data size. However, their methods do not explicitly take account of challenges in a harsh manufacturing workshop, especially using 2D Lidar. Also, the time consumption of pose initialization or localization failure recovery, which is important for manufacturing industry, is not sufficiently considered.

Fig. 2 shows a typical manufacturing workshop. In this environment, Lidar-based localization faces three serious challenges: rough and oily ground, dynamic obstacles and the convex plate of a CNC. Specifically, the rough and oily ground may cause wheel slip so that the accuracy of encoder based odometry is poor. Dynamic obstacles such as moving workers always lead to occlusion of a Lidar sensor. Moreover, due to the changing roll angle of a mobile manipulator on the rough ground, the two-layer structure of the convex plate will cause dramatic fluctuation of Lidar data.

In this paper, to explicitly address the abovementioned challenges, we develop a localization system by using hybrid maps, which allows accuracy and reliable localization in harsh manufacturing workshop. The developed localization system has been successfully applied to our self-developed 4WS4WD mobile manipulator. An average positioning error of 0.005 m/0.111 deg is obtained through two weeks of loading and unloading tasks. In our work, the robot localization technology is transferred from laboratory experiments to real-world applications and the contributions are as follows:

1) A localization system is developed using three different hybrid maps including hybrid grid map, MLFs and hybrid point map. Those maps are designed to model the harsh manufacturing environment in detail and to improve the performance of localization.
2) For reliable and fast pose initialization, the convex hull sampling(CS) based BnB localization is proposed through three strategies: the CS strategy to sparse dense Lidar data, MLFs based BnB search strategy for efficiency global search and multi-times localization strategy to improve reliability.
3) For accuracy and robust pose tracking under the situation of large odometry error and dramatic fluctuations of Lidar data, a localization method referred to as branch and bound adaptive Monte Carlo localization (BnB-AMCL) is proposed
4) For mitigating the influence of dynamic obstacles, a distance filter improved by unscented transform is proposed.
5) Experiments in a harsh manufacturing workshop are carried out to verify the performance of the proposed localization system.

This paper is organized as follows: the next section presents related works. The overview of the localization system is presented in Section III. The methods of constructing hybrid maps are introduced in Section IV. Our global localization method and BnB-AMCL are introduced in Section V and Section VI respectively. Different experiments in a real manufacturing workshop are presented in Section VII, followed by conclusions in Section VIII.

II. RELATED WORK

A. SENSORS FOR ROBOT LOCALIZATION

Choosing the proper sensor is the foundation for accuracy and reliable localization. The commonly used sensors are visual sensors, wireless sensor networks, radio frequency identification (RFID), 2D/3D Lidar and so on. Visual sensors such as mono-camera [6] and RGB-D camera [7] can obtain abundant information for robot localization. However, they are sensitive to the light changes and the field of view variations, which makes them difficult to meet the reliability requirements of industry application. Wireless sensor networks [8] and radio frequency identification (RFID) [9] are robust to environment changes and require little computation so that they allow robust and fast localization. However, about 10cm accuracy can be achieved through such sensors and it is not accurate enough for robots operating in industrial environments. Due to accurate distance measurement over a wide range, 2D/3D Lidar based localization can meet both requirements of accuracy and reliability [10]–[12]. 3D Lidar can obtain more environment information than 2D Lidar and it can achieve localization in six degrees of freedom. However, at present, 3D Lidar is much expensive than 2D and more information requires more computational resources. Thus, 2D Lidar has obvious advantages over 3D Lidar for robot localization in indoor manufacturing workshop.

B. GLOBAL LOCALIZATION

Global localization is a procedure to estimate the initial pose of a robot or to re-localize a robot after localization failure. With a large number of particles, MCL is a convenient way to solve the global localization problem [13]. The drawbacks of this method are that it is time-consuming and its success rate is low. By finding correspondences between a set of sparse features detected in two occupancy grid maps, a map to map matching based global localization is designed by Blanco *et al.* [14]. Park *et al.* adopt the SVM method to train a place classifier which can be used for global localization. However, their method relies on reliable feature detection which is a tough job for Lidar data [15]. Liu *et al.* employ the BnB strategy for global point cloud registration [16]. Also, the same strategy is employed for loop closure and vehicle localization [17]–[19]. Each of those four methods can be regarded as a global localization method and their experiments show that the BnB strategy allows performing a fast global search while maintains a high success rate. However, their methods do not try to sparse the dense Lidar data for further improving efficiency and their BnB strategy cannot be seamlessly combined with AMCL [20], [21].

C. POSE TRACKING

After getting the initial pose, pose tracking is performed to incrementally track the pose of a mobile manipulator. A graph-based pose tracking method using sparse point feature and lane marking is proposed by Wu *et al.* [22]. Their approach performs with less than 0.50 m/0.41 deg localization error in an outdoor city environment. Biswas *et al.* apply the state space gradients to refine the localization result of MCL and a centimeter-level accuracy can be achieved by their method [23]. An accurate pose tracking method based on particle filter and scan matching is verified through a highly accurate motion capture system [24]. They are able to achieve average localization error and positioning error in a laboratory static environment within 3mm/0.06deg and 5 mm/0.15 deg respectively. By using the discrete Fourier transform, Vasiljević *et al.* develop an accuracy pose tracking method for autonomous warehousing in an industrial setting whose positioning accuracy reached 1.5 cm and 0.5 deg [3]. Although accuracy pose tracking can be achieved through those methods, however, without explicitly considering the negative impacts of a harsh manufacturing workshop, their localization method may be hard to apply to the self-localization of mobile manipulators.

D. MAPS FOR ROBOT LOCALIZATION

To improve the localization performance in complex environments, previous works have employed different forms of maps. Due to its concise form and abundant information, the occupancy grid map [25] is commonly used for robot localization. Tipaldi *et al.* develop the dynamic occupancy grid to deal with the semi-static object in a dynamic environment, in which an RBPF based localization method is employed to simultaneously locate a robot and update the dynamic occupancy grid [26]. In our previous work [27] a novel localization method is proposed for robot localization in ambiguous environments, which is based on a pre-built map named ambiguity grid map that models the ambiguous property of an environment. To realize robust localization in an oil and gas industrial environment, a pre-computed 3d likelihood field using hybrid octree is presented by Merriaux *et al.* and efficiency likelihood computation can be obtained through this map [28]. By employing multi-layer maps, Vasiljevic *et al.* achieve a robust forklifts localization in a dynamic industrial environment with the MCL method [29]. Those methods show that maps are very useful to enhance the localization performance in some special scenes. Therefore, it is a reason for this paper to explore hybrid maps to make up for the shortage of 2D Lidars in harsh manufacturing workshop and to improve localization accuracy and reliability.

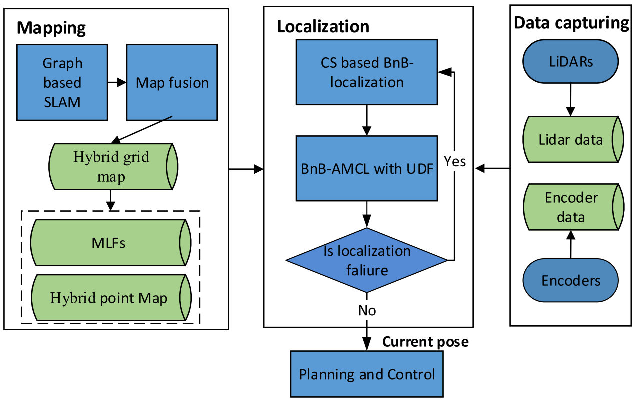


FIGURE 3. Localization system architecture.

III. OVERVIEW OF THE LOCALIZATION SYSTEM

The proposed localization system, as demonstrated in Fig. 3, mainly contains three modules: data capturing, mapping and localization, which integrally solves the localization problem in harsh manufacturing workshop. Each module of our localization system are briefly explained below:

1) **Data capturing module:** This module is used to read and preprocess the original sensor data from two Lidars and eight encoders.
2) **Mapping module:** Three maps including hybrid grid map, MLFs and hybrid point map are created in this module. Hybrid grid map models the two-layer structure of a manufacturing workshop, which can be used to deal with the dramatic fluctuation of Lidar data. MLFs are used for computing the match score and the upper bound during the BnB search while improving search efficiency. Also, this map can bridge the gap between BnB search and AMCL so that those two methods can be combined seamlessly. Hybrid point map is used for ICP based accuracy localization.
3) **Localization module:** CS based BnB localization is employed to fast and reliably obtain the initial pose of a

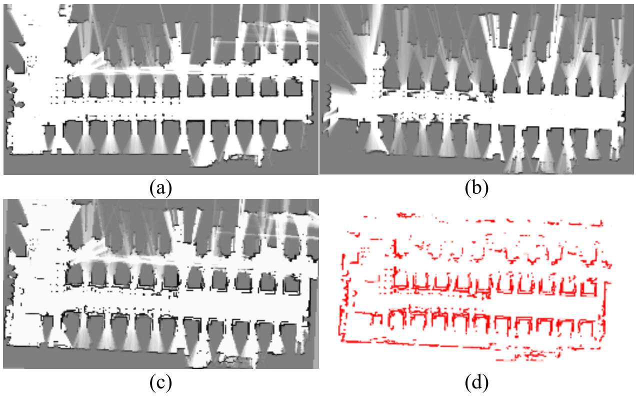


FIGURE 4. Combination of two occupancy grid maps at different height: (a) Occupancy grid map below standard height. (b) Occupancy grid map above standard height. (c) Hybrid grid map. (d) Hybrid point map.

robot. Subsequently, the BnB-AMCL with UDF is used to filter out dynamic obstacles and to track robot pose accurately. Pose tracking procedure is under the supervision of localization failure detection which using the match score produced by BnB-AMCL to detect distinct localization failure. Successful tracking results will be adopted by the planning and control module. Otherwise, the robot will stop for re-localization through our global localization method and then resume its work after locating itself.

IV. MAPPING

In this section, we will introduce the building process of the hybrid grid map. The hybrid point map can simply be created through binarization of the hybrid grid map and then converting its pixel coordinates to the Euclidean coordinates according to its grid resolution. The detailed building process of the MLFs is presented in Section V.

The standard Lidar installation height for our mobile manipulator is designed to meet the requirement of mechanical structure and electrical arrangement of the mobile platform. Then, the hybrid grid map is a combination of two occupancy grid maps at different heights which are below and above the standard Lidar installation height. Concretely, the building process of the hybrid grid map is as follows: 1) By mounting a Lidar on a height of 5cm higher and lower than the standard installation height respectively and manually steering the robot traversing through a workshop, two occupancy grid maps of different heights are built through graph-based SLAM method [17]. The resulting occupancy grid maps are shown in Fig. 4 (a) and (b); 2) The coordinate systems of these two maps are inconsistent. Therefore, we use the BnB based global localization method presented in Section V to obtain a coarse coordinate transformation of those two maps; 3) The two occupancy grid maps are then converted into point maps. Afterward, by using the coarse coordinate transformation as the initial alignment, the refined coordinated transformation can be computed through point cloud registration such as ICP. It should be noted that one point cloud registration may not be enough to get an ideal alignment. In this situation, multi- times of point cloud registration should be performed with different initial coordinate transformations which are close to the coarse coordinated transformation. After alignment, those two grid maps have a consistent coordinate system; 4) Finally, the occupancy probability of each grid map should be updated. More specifically, we define $\boldsymbol{p}$ as the pixel coordinate of a grid map, $O_h(\boldsymbol{p})$ as the hybrid gird map and $O_1(\boldsymbol{p})$, $O_2(\boldsymbol{p})$ as the occupancy grid map at the lower height and the higher height respectively. Then $O_h(\boldsymbol{p})$ can be calculated according to Table 1. The result hybrid grid map is shown in Fig. 4 (c), and the corresponding hybrid point map is shown in Fig. 4 (d).

TABLE 1. Occupancy probability calculation.

$O_1(\boldsymbol{p})$	$O_2(\boldsymbol{p})$	$O_h(\boldsymbol{p})$
< 0.4	*anyvalue*	$O_1(\boldsymbol{p})$
anyvalue	< 0.4	$O_2(\boldsymbol{p})$
> 0.7	≥ 0.4	$O_1(\boldsymbol{p})$
≥ 0.4	> 0.7	$O_2(\boldsymbol{p})$
else		$\frac{O_1(p)+O_2(p)}{2}$

V. CS BASED BnB-LOCALIZATION

A reliable global localization requires a comprehensive search in the solution space constrained by an occupancy grid map and the original Lidar data for global localization is dense. Thus, those two issues make global localization a time-consumption task. In this paper, for a fast and reliable global localization, we employ three strategies: the CS strategy to sparse Lidar data, a MLFs based BnB search strategy to efficiently and reliably find the global optimal solution and a multi-times localization strategy to improve reliability.

A. CS STRATEGY

CS is achieved by extracting convex hull from the original Lidar data [30], followed by a random sampling procedure to preserve more Lidar data. A convex hull of a given point set can be defined as the smallest convex polygon containing all points in the point set. There are two reasons to employ convex hull: 1) Convex hull, to some extent, characterize the shape property of a point set; 2) Relative small distance measurements usually arise from dynamic obstacles. However, those points belonging to a convex hull correspond to large distance measurements which are more likely caused by static objects such as walls and machine tools.

By using CS, the amount of the original Lidar data can be significantly decreased, however, the concave property of Lidar data is disregarded. To address this problem, some extra points are randomly sampled from the original points. To balance the influence of randomly sampled points, the number of

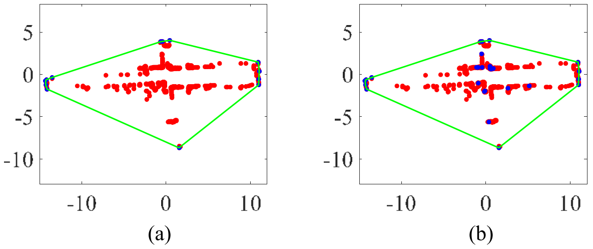


FIGURE 5. Schematic diagram of CS: Red points and blue points represent the original Lidar data and the sampled data by using CS respectively. The green polygon represents the convex hull of the original data. (a) Convex hull extracted from the original Lidar data. (b) Convex hull and random samples.

such points is equal to the number of the convex hull points. An example of CS can be seen from Fig. 5.

B. MLFs BASED BnB SEARCH

Global localization can be formulated as an optimal match problem which aims at finding a pose x that maximum a match score:

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}\in\mathcal{W}} \sum_{n=1}^{N_{Lidar}} M(\boldsymbol{T}_x \boldsymbol{z}^{[n]}) \tag{1}$$

where $\boldsymbol{T}_x$ denotes a transform matrix derived from a pose $\boldsymbol{x} \triangleq (x_x, y_x, \theta_x)^T$ and $z^{[n]} \triangleq (x_{z^{[n]}}, y_{z^{[n]}})^T$ denotes the n th point from the sampled Lidar data. N_{Lidar} is the number of Lidar data. More specifically,

$$\boldsymbol{T}_x \boldsymbol{z}^{[n]} = \begin{pmatrix} \cos\theta_x & -\sin\theta_x \\ \sin\theta_x & \cos\theta_x \end{pmatrix} \begin{pmatrix} x_{z^{[n]}} \\ y_{z^{[n]}} \end{pmatrix} + \begin{pmatrix} x_x \\ y_x \end{pmatrix} \tag{2}$$

M is a map for computing the match score, i.e.: M: $r\mathbb{Z} \times r\mathbb{Z} \rightarrow \mathbb{R}$, where r is the resolution of M. In our implementation, we use likelihood fields as M.

$\mathcal{W}$ is the discrete solution space which contains all pixel-accuracy pose in an occupancy grid map, i.e.:

$$\begin{aligned} \mathcal{W} = &\{x_{\min} + ir : i \in \mathbb{N}, x_{\min} + ir \leq x_{\max}\} \\ &\times \{y_{\min} + jr : j \in \mathbb{N}, y_{\min} + jr \leq y_{\max}\} \\ &\times \{-\pi + k\delta_\theta : k \in \mathbb{N}, -\pi + k\delta_\theta < \pi\} \end{aligned} \tag{3}$$

where $\times$ represents the Cartesian product and i, j, k denote index. $x_{\min}$, $y_{\min}$, $x_{\max}$, $y_{\max}$ are the bounding box of the map M. δ_θ is the angular step for angular search. Given a robot pose x, the match score $\sum_{k=1}^{N_{Lidar}} M(\boldsymbol{T}_x \boldsymbol{z}^{[n]})$ tells us the match quality between sampled Lidar data and a map M.

We employ the BnB strategy to solve the above optimal match problem, where the node selection method, branch method and upper bound computation method need to be specified. Similar to the implementation of BnB method in [17], we employ a depth-first search to select node and quadtree to divide the X-Y dimension while the angular dimension is searched exhaustively with the angular step δ_θ. As shown in Fig. 6, given a fixed angle, the X-Y solution space is recursively divided into four sub-spaces until a sub-space only consists of one possible pose. After each subdividing, the upper bound of each sub-space, which is a value bigger than or equal to the maximum match score of those poses in that sub-space is computed and the sub-space with the highest upper bound is selected for the next subdividing. Moreover, if the upper bound of the possible pose, which is also the match score of that pose, is higher than the upper bound of other sub-space, those sub-spaces are pruned and the possible pose is the optimal pose. Otherwise, the sub-space with the highest upper bound is continued searched with the depth-first search.

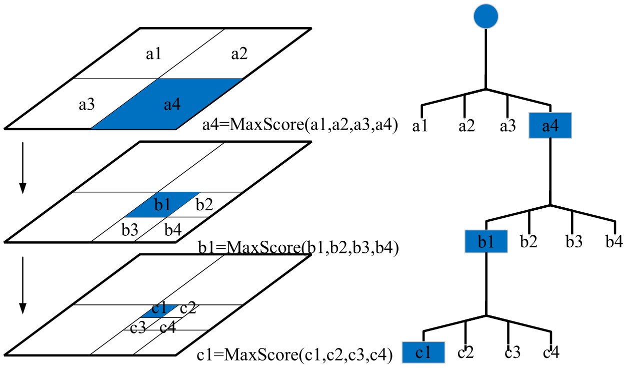


FIGURE 6. Schematic diagram of BnB based search: The left part shows the recursive division of a search space using the depth-first search. The right part shows the corresponding search tree. The blue regions indicate the sub-spaces with the highest upper bound.

The same equation $\sum_{n=1}^{N_{Lidar}} M(\boldsymbol{T}_x \boldsymbol{z}^{[n]})$ is used for upper bound computation, the difference is that at this time MLFs are used as M. MLFs are a generalization of likelihood fields which is a kind of grid map made up by cells and is developed to compute the observation likelihood for robot localization [31]. The advantage of MLFs for BnB based global localization is that they make localization robust to sensor noise and cluttered environment. In addition, they allow a relatively large angular step for angular exhaustive search. Furthermore, by using likelihood fields to compute the match score, the BnB search and AMCL can be combined seamlessly which will be detailed described in the next section. The MLFs can be expressed as:

$$M_{LF}^0(\xi) = \alpha p_{rand} + (1-\alpha)\frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(\|\xi - \xi_{nearest}\|)^2}{2\sigma^2}} \tag{4}$$

$$\begin{aligned} M_{LF}^h(\xi) = &\max(\{M_{LF}^0(i) : \\ &i \in \{(i_x, i_y) \in \mathbb{Z}^2 : \begin{pmatrix} \xi_x \leq i_x \leq \xi_x + 2^h - 1 \\ \xi_y \leq i_y \leq \xi_y + 2^h - 1 \end{pmatrix}\}\}) \end{aligned} \tag{5}$$

where $M_{LF}^0(\xi)$ represents the likelihood fields of a hybrid grid map and also it is the likelihood fields with the highest resolution, whose resolution equals to the resolution of the hybrid grid map. ξ indicates a cell in the 2D grid map, $\xi_{nearest}$ indicates the occupied cell nearest to ξ in the hybrid grid map,

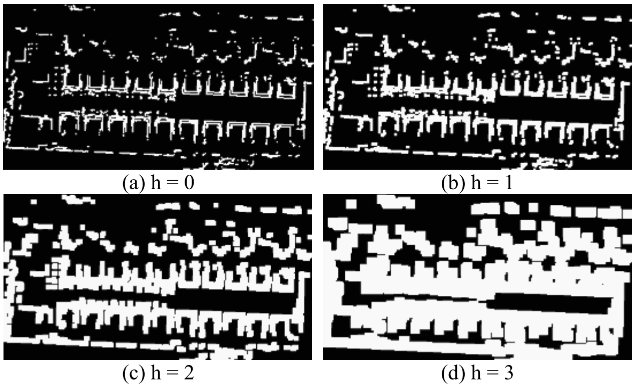

(a) h = 0
(b) h = 1
(c) h = 2
(d) h = 3

FIGURE 7. An example of MLFs.

σ indicates the distance standard deviation of a Lidar and α is a weight parameter. Equation (4) demonstrates that a cell value of a likelihood field is a weighted sum of two distribution, one of which is a uniform distribution parameterized by p_{rand} and the other one is a Gaussian distribution.

To define an integer h greater than zero as the level of resolution, the MLFs is represented by M_{LF}^{h} and can be built according to Equation (5). In detail, the value of a cell ξ for M_{LF}^{h} is the maximum cell value of a square region in M_{LF}^{0}, where the top left corner of the square region is at ξ and its length equals $2^h - 1$ which indicates the resolution at the level h. In fact, the size of the square region indicates the size of a sub-space during the BnB search. And the MLFs are look-up tables for the value that is bigger than or equal to the maximum match score of a sub-space, and this value is the upper bound of such sub-space. An example of MLFs is shown in Fig. 7.

Using the law of cosine, the angular step can be computed according to (6) and (7).

$$d_{\max} = \max_{n=1,\ldots,N_{Lidar}} \left\| z^{[n]} \right\| \tag{6}$$

$$\delta_\theta = \arccos(1 - \frac{2\sigma^2}{d_{\max}^2}) \tag{7}$$

As we can see, the angular step is selected that the maximum distance measurement does not move more than the distance standard error of a Lidar. This selection method will give us a relatively small angular step and ensure a high success rate for localization.

C. PSEUDO-CODE OF OUR GLOBAL LOCALIZATION METHOD

Besides CS and MLFs based BnB search, a multi-times localization strategy is employed to further improve localization reliability. The complete pseudo-code of our global localization method is shown in Table 2. In line 2, the point set C which is the convex hull of the original point set P is extracted. Line 4 to 10 shows a loop that implements the multi-times localization strategy. This loop will be break if the current match score is higher than score_reliable or reaches the maximum loop count max_gloc_count. At each localization, we first obtain D through combining the randomly sampled point set R and C. Then BnB search is employed in step 7. One important thing to note is that a parameter named minmum_score is induced for the BnB search. This parameter constrains the minimum match score. Those sub-spaces whose upper bound are lower than this parameter are pruned. It is not necessary for global localization. However, this parameter will benefit our pose tracking method described below. In line 8, the current score is computed where $\{z_p^{[n]}\}$ are those points from the original point set P and then it is used to decide whether or not to drop out of the loop.

TABLE 2. CS based BnB-Localizaiton.

```
Input: Multi-resolution likelihood field $\{<M_{LF}^{h}>_{h=0}^{4}\}$, Lidar observation
        data $z$ , minmum_score
Output: robot pose $x$ and $score$
1:  a point set $P$ is obtained from transforming the observation
    data $z$ expressed in polar coordinates to Cartesian coordinates
2:  extract convex hull of $P$ to get point set $C$
3:  $count \leftarrow 0$
4:  while cur_score < score_reliable & count < max_gloc_count do
5:    sample point set $R$ from $(P\text{-}C)$ , the size of $R$ is equal to the
      size of $C$
6:    $D = C \cup R$
7:    $x = branchAndBound(D, \{<M_{LF}^{h}>_{h=0}^{4}\}, \text{minmum\_score})$
8:    $\text{cur\_score} = \sum_{n=1}^{N_{Lidar}} M_{LF}^{0}(\boldsymbol{T}_x \boldsymbol{z}_P^{[n]})$
9:    count ← count+1
10: end
11: return $x$ and cur_score
```

VI. BnB-AMCL WITH UDF

The aim of the BnB-AMCL with UDF localization method is to address three issues in harsh manufacturing workshop: 1) the inaccuracy odometry caused by the rough and oily ground; 2) dramatic fluctuation of Lidar measurements caused by two-layer structure and rough ground; 3) Lidar occlusion caused by dynamic obstacles. In our implementation, we employed the AMCL method as the basic localization method, meanwhile, some novel strategies are adopted to improve AMCL so that it can meet the requirements in harsh manufacturing workshop. The odometry is computed by the data from 8 encoders and is used for the motion model of AMCL. In this paper, we mainly focus on the overall localization process, the detail of the odometry computation is out of the scope. Researches on encoder odometers have been thoroughly studied in [32] and [33].

A. BnB-AMCL

Given a prior occupancy grid map $\boldsymbol{m}_g$, robot localization can be considered as a probabilistic reasoning problem:

$$Bel(\boldsymbol{x}_t) = P(\boldsymbol{x}_t | z_{1:t}, \boldsymbol{u}_{1:t}, \boldsymbol{m}_g) \tag{8}$$

where $\boldsymbol{x}_t$ denotes the robot pose at a time t, $z_{1:t}$ is the sequence of observations, $\boldsymbol{u}_{1:t}$ denotes a sequence of motion control commands or odometry measurements.

Based on Markov independence of the odometry and Markov independence of the observation [34], (8) can be rewritten as:

$$Bel(\boldsymbol{x}_t) = \eta P(z_t|\boldsymbol{x}_t, \boldsymbol{m}_g) \cdot \int_{\boldsymbol{x}_{t-1}} P(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{u}_t) Bel(\boldsymbol{x}_{t-1}) d\boldsymbol{x}_{t-1} \tag{9}$$

where η is a normalization coefficient, $P(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)$ and $P(z_t|\boldsymbol{x}_t, \boldsymbol{m}_g)$ correspond to the motion model and observation model, respectively.

In general, the pose which maximum the posterior $Bel(\boldsymbol{x}_t)$ is selected as the current pose estimation, which is expressed as:

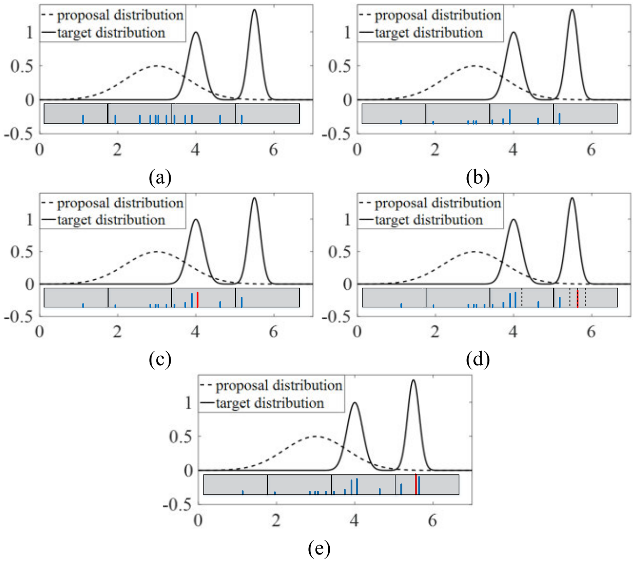
$$\hat{\boldsymbol{x}}_t^{MAP} = \arg\max_{\boldsymbol{x}_t} Bel(\boldsymbol{x}_t) \tag{10}$$

By using a set of weighted samples $\{< \omega_t^{[i]}, \boldsymbol{x}_t^{[i]} >_{i=1}^{N_t}\}$, also referred to as particles to represent probability distribution, Monte Carlo localization (MCL) [25] is an approximate but efficient way to solve the maximum posterior estimation of (10). The key technique of MCL is to firstly sample from a so-called proposal distribution which allows efficiently sampling. Then, the weight of each particle is computed through a likelihood function. After that, the current pose posterior distribution $Bel(\boldsymbol{x}_t)$, which also referred as to target distribution, can be expressed by those weighted particles and the particle with the highest weight is select as the current pose estimation. Finally, resampling is performed to obtain particles which have the same weight. For standard MCL, the motion model $P(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)$ is chosen as the proposal distribution and the likelihood fields which is an instantiation of the observation model $P(z_t|\boldsymbol{x}_t, \boldsymbol{m}_g)$ is adopted to compute the particle weight. To adaptive the particle set size for further efficiency, the AMCL method is developed by KLD-sampling. This method shows that the more the uncertainty of a motion process, the more the number of particles is required, and vice versa. To implement the KLD-sampling, some bins with a fixed size will be defined. Once a particle falls into a new bin during the sampling process, the number of particles is updated according to

$$n = \frac{m-1}{2\varepsilon} \left\{ 1 - \frac{2}{9(m-1)} + \sqrt{\frac{2}{9(m-1)}} z_{1-\delta} \right\}^3 \tag{11}$$

where ε and δ are pre-specified error bound threshold, $z_{1-\delta}$ is the upper quantile of the standard normal distribution. m is the number of bins.

In harsh manufacturing workshop, inaccuracy odometry will cause a relative dispersive proposal distribution. Meanwhile, the target distribution may be multimodal because of the two-layer structure. Under these circumstances, AMCL always gives a poor pose estimation and requires a lot of particles. This situation is shown in Fig. 8 (a) and (b). Fig. 8 (a) shows those particles sampled from proposal distribution. Those particles are widespread and a little of them particle is near the location with the largest posterior. In Fig. 8 (b), the weight of each particle is presented, which is computed by using observation likelihood function. We can clearly see that the particle with the largest weight is far from the location with the largest posterior.



(a) (b) (c) (d) (e)

FIGURE 8. The process of BnB-AMCL: Dashed line and solid line represent proposal distribution and target distribution respectively. Each stick represents a particle and its length indicates the weight of a particle. Red sticks represent particles produced by BnB search or ICP based scan match. Gray boxes represent those bins produced by KLDsampling. (a) Prediction step. (b) Update step. (c) ICP step. (d) BnB step. (e) re-ICP step.

As we know from Section IV, the BnB search can be used to obtain a global optimal pose, therefore we combine BnB search and AMCL to develop a pose tracking method called BnB-AMCL to solve the abovementioned issues. In fact, there are relations between those two methods. Those bins generated during KLD-sampling can be regarded as a search space of BnB. Furthermore, by using likelihood fields, the computation method for match score is the same method for weight computation. That is if we use the maximum weight from AMCL as the minmum_score, BnB search will give us a global optimal pose whose weight is higher than the maximum weight from AMCL and inefficient searches are avoided because those sub-spaces whose upper bound are lower than minmum_score are pruned.

To obtain a more accuracy pose estimation in BnB-AMCL, we employed ICP to match the current Lidar data with the hybrid point map. Specifically, BnB-AMCL has three more steps after weight computation: 1) find pose $\boldsymbol{x}_t^a$ with the maximum weight and obtain a particle $< \omega_t^b, \boldsymbol{x}_t^b >$ (Fig. 8 (c)) by performing ICP which uses $\boldsymbol{x}_t^a$ as the initial alignment; 2) take ω_t^b as the minmum_score and the bins generated from the KLD-sampling process as the initial search space, a particle $< \omega_t^c, \boldsymbol{x}_t^c >$ (Fig. 8 (d)) is obtained through BnB search; 3) ICP is again performed by using $\boldsymbol{x}_t^c$. as the initial alignment to get the final pose estimation $< \omega_t^d, \boldsymbol{x}_t^d >$ (Fig. 8 (e)).

Once the ICP based matching performs successfully, it is obvious to know that $\omega_t^d \geq \omega_t^c \geq \omega_t^b \geq \omega_t^a$. Besides, with the global optimality of the BnB search, $\boldsymbol{x}_t^d$ corresponds to the pose with the largest posterior among the search space generated from KLD-sampling. Therefore, a more reliable and accuracy pose estimation can be acquired by our method. One important thing to note that if we choose the bin size the same as the lowest resolution of the MLFs, only fewer particles is enough for BnB-AMCL. This is because it is a relative large bin size and a wide initial search space for the BnB search will be generated with such size.

B. UDF

Dynamic obstacles may produce distance measurements that are shorter than expected. Those relative short measurements can be filtered through the distance filter [35]. To effectively integrated distance filter with BnB-AMCL, unscented transformation is adopted to improve the original distance filter, which leads to UDF.

The probability that a distance measurement $d^{[i]}$ at index i is shorter than expected is expressed as:

$$p_{short}(d^{[i]}) = \int p_{short}(d^{[i]}|\boldsymbol{x})p(\boldsymbol{x})dx \tag{12}$$

where $\boldsymbol{x}$ denotes robot pose and $p(\boldsymbol{x})$ is the distribution of robot pose at the current time. Given $\boldsymbol{x}$ and a grid map, the expected distance measurement $d_\mu^{[i]}$ at index $\boldsymbol{i}$ can be obtained through the raycasting algorithm. If $p_{short}(d^{[i]})$ is greater than a predefined threshold t_b, we think $d^{[i]}$ is caused by moving workers and such measurement will be ignored during localization. $p_{short}(d^{[i]}|\boldsymbol{x})$ can be expressed as a cumulative probability of a Gaussian distribution $N(d_\mu^{[i]}, \sigma_d)$:

$$p_{short}(d^{[i]}|\boldsymbol{x}) = \int\limits_{d_j > d^{[i]}} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(d_j - d_\mu^{[i]})^2}{2\sigma^2}} dd_j \tag{13}$$

By using the look-up table, the cumulative probability of (13) is easy to compute. In (12), $p(\boldsymbol{x})$ is unknown. Perhaps, the goal of localization is to compute $p(\boldsymbol{x})$. Using those particles produced in the prediction step of BnB-AMCL is an approximate and feasible way to represent $p(\boldsymbol{x})$. However, each particle requires one raycasting operation. And a lot of computation may be required with too many particles. Hence, we adopt unscented transformation in three dimensions to reduce the number of particles to 7 to represent $p(\boldsymbol{x})$. More specifically, based on the unscented transformation, those seven particle and their corresponding weight are expressed as:

$$\begin{aligned}
\boldsymbol{x}_U^{[0]} &= \bar{\boldsymbol{x}}_{prior} \quad \omega_U^{[0]} = \kappa/(n_x + \kappa)\ i = 0 \\
\boldsymbol{x}_U^{[i]} &= \bar{\boldsymbol{x}}_{prior} + \left(\sqrt{(n_x + \kappa)\boldsymbol{P}_{prior}}\right)_i \\
&\quad \times \omega_U^{[i]} = 1/\{2(n_x + \kappa)\} \quad i = 1, \ldots, n_x \\
\boldsymbol{x}_U^{[i]} &= \bar{\boldsymbol{x}}_{prior} - \left(\sqrt{(n_x + \kappa)\boldsymbol{P}_{prior}}\right)_{i-n_x} \\
&\quad \times \omega_U^{[i]} = 1/\{2(n_x + \kappa)\} \quad i = n_x + 1, \ldots, 2n_x
\end{aligned} \tag{14}$$

TABLE 3. Pseudo-code for BnB-AMCL with UDF.

```
Input: $S_{t-1} = \{<\omega_{t-1}^i, \boldsymbol{x}_{t-1}^i>_{i=1}^{N_{t-1}}\}$, $\boldsymbol{z}_t$, $\boldsymbol{u}_t$, $\boldsymbol{u}_{t\text{-}1}$, $\{<M_{LF}^h>_{h=0}^4\}$, $M_p$
Output: current pose estimation $\boldsymbol{x}_t$
1:  $S_t \leftarrow \varnothing$, $\eta \leftarrow 0$, $k = 0$, $B \leftarrow \varnothing$, $N_t \leftarrow 0$
2:  compute odometry $\{\Delta x, \Delta y, \Delta\varphi\}$ according to encoder reading
    $\boldsymbol{u}_{t-1}$ and $\boldsymbol{u}_t$
3:  do
4:      resampling an index $j$ from $S_{t-1}$
5:      create motion model $P(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}^{[j]}, \{\Delta x, \Delta y, \Delta\varphi\})$
6:      sample $\boldsymbol{x}_t^{[n]}$ from $P(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}^{[j]}, \{\Delta x, \Delta y, \Delta\varphi\})$
7:      $S_t \leftarrow S_t \cup \{<1, \boldsymbol{x}_t^{[n]}>\}$
8:      compute bin position $b$ according to $\boldsymbol{x}_t^{[n]}$
9:      if ( $B$ not contains $b$ ) then
10:         $k \leftarrow k + 1$
11:         insert $b$ into $B$
12:         update $N_t$ according to (11)
13:     endif
14:     $n \leftarrow n + 1$
15: while( $n < N_t$ and $n < n_{\min}$ )
16: Using $S_t$ to represent pose belief , $\boldsymbol{z}_t$ is filtered by UDF.
17: for $i \leftarrow 1, \ldots, n$ do
18:     $\omega_t^{[n]} \leftarrow P(\boldsymbol{z}_t \mid \boldsymbol{x}_t^{[n]})$ , $\eta = \eta + \omega_t^{[n]}$
19:     $S_t[i] \leftarrow <\omega_t^n, \boldsymbol{x}_t^n>$
20: endfor
21: find $<\omega_t^a, \boldsymbol{x}_t^a>$ with the largest weight from $S_t$
22: $<\omega_t^b, \boldsymbol{x}_t^b> \leftarrow ICP(\boldsymbol{x}_t^a, \boldsymbol{z}_t, M_p)$
23: $<\omega_t^c, \boldsymbol{x}_t^c> \leftarrow BranchAndBound(\boldsymbol{z}_t, B, \{<M_{LF}^h>_{h=0}^4\}, \omega_t^b)$ ,
24: $<\omega_t^d, \boldsymbol{x}_t^d> \leftarrow ICP(\boldsymbol{x}_t^c, \boldsymbol{z}_t, M_p)$
25: $S_t \leftarrow S_t \cup \{<\omega_t^b, \boldsymbol{x}_t^b>, <\omega_t^c, \boldsymbol{x}_t^c>, <\omega_t^d, \boldsymbol{x}_t^d>\}$ , $n \leftarrow n + 3$ ,
    $\eta \leftarrow \eta + \omega_t^b + \omega_t^c + \omega_t^d$
26:  for $i \leftarrow 1, \ldots, n$ do
27:     $\omega_t^{[i]} \leftarrow \omega_t^{[i]} / \eta$
28: endfor
29: return $\boldsymbol{x}_t^d$
```

where $\boldsymbol{x}_U^{[i]}$ and $\omega_U^{[i]}$ represent the i th particles produced by unscented transformation and its corresponding weight. n_x is the dimension of $\boldsymbol{x}$, which is set to 3. $\bar{\boldsymbol{x}}_{prior}$ and $\boldsymbol{P}_{prior}$ denote the mean and covariance matrix of those particles produced by the prediction step of BnB-AMCL, and κ represents a scaling parameter, which is set to 1 in our implementation.

Then (12) can be rewritten as:

$$p_{short}(d_i) = \sum_{i=1}^{2n_x+1} p_{short}(d_i|\boldsymbol{x}_U^{[i]})\omega_U^{[i]} \tag{15}$$

By using (15), the improved distance filter, referred to as UDF, can be efficiently integrated into BnB-AMCL to mitigate the influence of dynamic obstacles.

C. PSEUDO-CODE OF OUR POSE TRACKING METHOD

The pseudo-code for BnB-AMCL with UDF is shown in Table 3. In step 2, odometry is computed through encoder readings. The prediction step is shown from step 5 to step 7. By using KLD-sampling, the size of particles is update from

FIGURE 9. The harsh manufacturing workshop for our experiments.

TABLE 4. Parameters for our experiments.

Name	Symbol	Value
Resolution of the hybrid grid map	r	0.05m
Lowest resolution level for MLFs	$h_{\max}$	4
Distance standard error of Lidar	σ	0.03m
Weight in (4)	α	0.05
Probability density for the uniform distribution in (4)	p_{rand}	0.00005
Minimum match score for localization failure detection.	minmum_score	0.3 times of the maximum score (the maximum score can be obtained through experiments which is 0.51 in our experiments)
Error threshold 1 for (11)	ε	0.03
Error threshold 2 for (11)	δ	0.01
Minimum number of particles for BnB-AMCL	$n_{\min}$	100
Threshold for UDF	t_b	0.9

step 8 to step 13, also bin set B is generated. $n_{\min}$ is a parameter which constrains the minimum particle number. In step 16, the UDF is adopted to filter Lidar data. Then update step is performed from step 17 to step 20. Step 21 to 25 shows the ICP and BnB step, where bin set B is used as the initial search space for BnB step. After that, all weights are normalized.

VII. EXPERIMENTS AND RESULTS

To comprehensively evaluate the performance of the proposed localization system, our experiments are carried out from two aspects: global localization and pose tracking. We conduct all experiments in a real manufacturing workshop shown in Fig. 9 using our self-developed 4WS4WD mobile manipulator. The hybrid maps for this workshop are created before experiments, which can be seen in Fig. 4 and Fig. 7. Some parameters for our experiments are shown in Table 4.

A. GLOBAL LOCALIZATION EXPERIMENTS

Offline Lidar data from 11 different stations are collected for our global localization experiments. At each station, we record 500 frames of Lidar data so that 500 times of global localization can be performed at one station. Those stations indicate pre-taught positions where the robot performs loading and unloading tasks or automatic charging, as shown in Fig. 10. In these experiments, we mainly focus on the success rate and average running time. Specifically, we assume a global localization is successful if its distance error is smaller than 0.5 m and its orientation error is smaller than 5 deg.

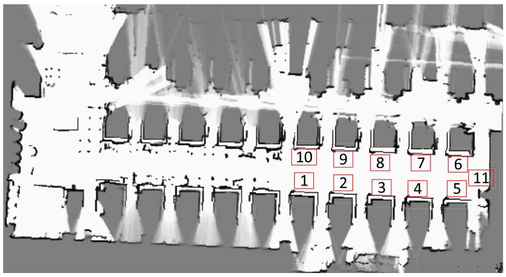


FIGURE 10. Selected stations for experiments: All stations indicate pretaught positions for loading and unloading workpieces or automatic charging.

In our global localization experiments, we compare three methods: branch and bound localization (BnB-localization) with all data, random sampling (RS) based BnB-localization and CS based BnB-localization. For random sampling, the sample set size is set to be consistent with the convex set based sampling for comparison. It is worth mentioning that MCL and map to map matching based global localization [14] as common global localization algorithms are also verified in our experimental environment. But the above methods require 1s-30s and their success rate are low, so their results are not detailed in our global localization experiments.

Fig. 11 shows the global localization results. In terms of the success rate, which is shown in Fig. 11(a), CSBnB-localization and BnB-localization with all Lidar data have similar performance, while RSBnB-localization gets the worst result among three methods. Interestingly, at station 7, the CSBnB-localization method gets a much higher success rate than that of BnB-localization with all Lidar data. For a clear explanation, an example of localization results at station 7 is shown in Fig. 12. In this example, CSBnB-localization achieves a success localization while others are not. From the original Lidar data (Fig. 12 (b)) at station 7, we can see that only a few measurements correspond to the right wall. Hence it is hard to determine the location in the horizontal direction. However, as shown in Fig. 12(c), through CS, the proportion of those measurements corresponding to the right wall is increased. In addition, more dynamic obstacles are removed and more useful information is reserved by CS. Therefore, compared to the other two methods, CSBnB-localization tends to achieve a successful localization. Fig. 11(b) shows the results of the average running time. As we can see sampling based localization method is much faster than using all Lidar data. Further, at each station, the average running time of the proposed CSBnB-localization is less than RSBnB-localization. This is because those samples produced

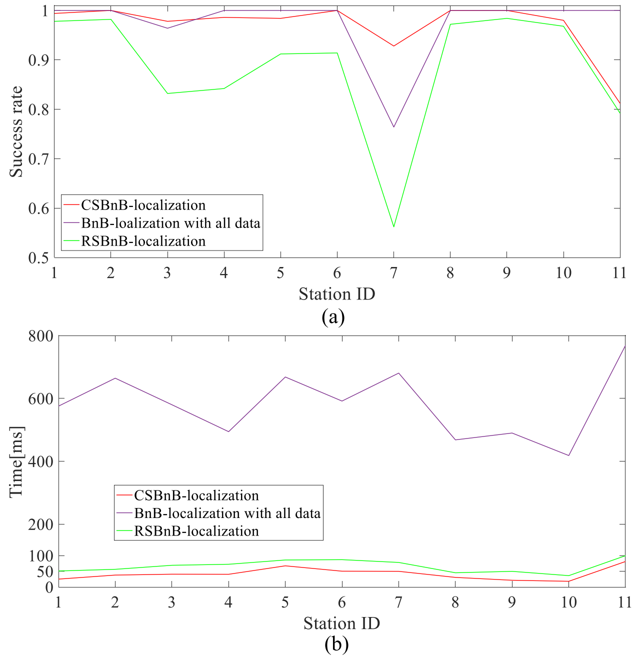


FIGURE 11. Localization performance of different global localization methods: (a) Success rate. (b) Average running time.

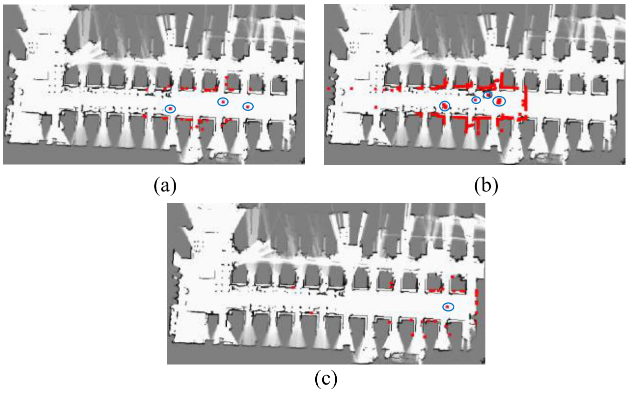


FIGURE 12. Registered Lidar data according to different global localization results: Red points represent Lidar measurements. Those measurements caused by dynamic obstacles are shown in blue circles. Both (a) RSBnB-localization and (b) BnB-localization with all data result a wrong registration, however(c) CSBnB-localization get a right localization result. Compared with (c), (a) and (b) involve more undesirable measurements caused by the dynamic obstacles. In addition, (c) reserves the shape property of the original Lidar data.

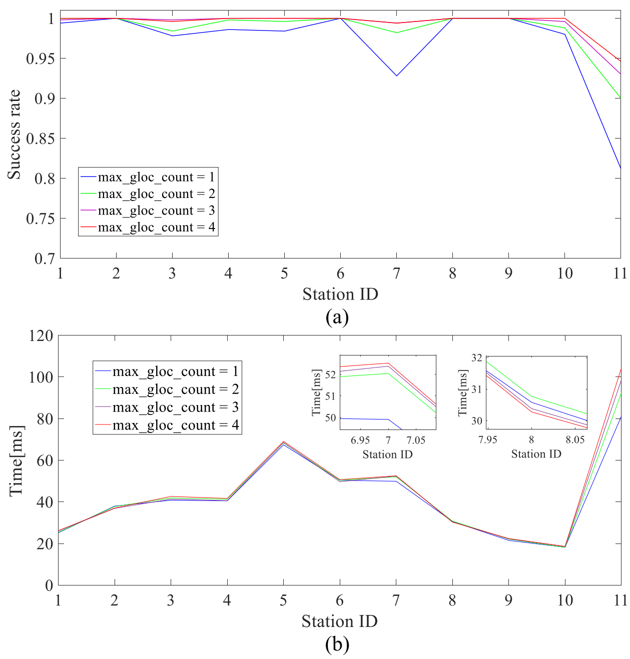


FIGURE 13. Global localization performance of CSBnB-localization with different max_gloc_count: (a) Success rate. (b) Average running time.

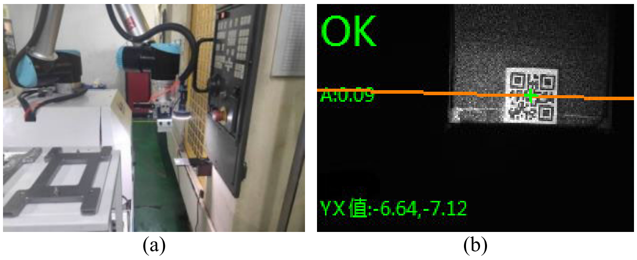
by CS can better express the shape of the original Lidar data and can result in more pruned branches during the BnB search.

It only takes 10 to 60 ms to run a CSBnB-localization, which make it possible to perform multiple times of localization to improve reliability while ensuring fastness. Therefore, we carry out experiments to show the global localization performance with different repetition times, which is denoted by parameter max_gloc_count. From Table 2, we know that once a good localization result is obtained the localization process will terminate or it will max_gloc_count. The max_gloc_count is set from 1 to 4 in our experiment and the results are shown in Fig. 13. It can be clearly seen that our global localization method has 100% success rate at most stations when max_gloc_count equals 4 and there is a little increment for the average localization time as max_gloc_count increasing. In addition, some stations such as station 8 get 100% success rate through only one time of localization. At those stations, the average localization time is not increased because of the early terminated localization process.

Through the global localization experiments, it has been verified that CSBnB-localization using MLFs has a very outstanding performance for balancing reliability and efficiency, which can be used for pose initialization and rapid localization failure recovery in an industrial environment.

B. POSE TRACKING EXPERIMENTS

The effectiveness of the proposed pose tracking method is verified from two aspects: localization accuracy and positioning accuracy. Localization accuracy is only related to the localization system while in addition to the localization system, positioning accuracy is also affected by the mechanical structure and control strategy of the mobile platform. Pose tracking experiments are all performed in a real manufacturing scenario which means more than 3 workers are moving during the experiment. For localization accuracy, we compare four localization methods including AMCL, AMCL with ICP, BnB-AMCL and BnB-AMCL with UDF. In addition, we also verify the performance of our localization method using hybrid maps and single layer maps. Given a station, its localization accuracy is evaluated through localization deviation and standard deviation of 500 localization results, which

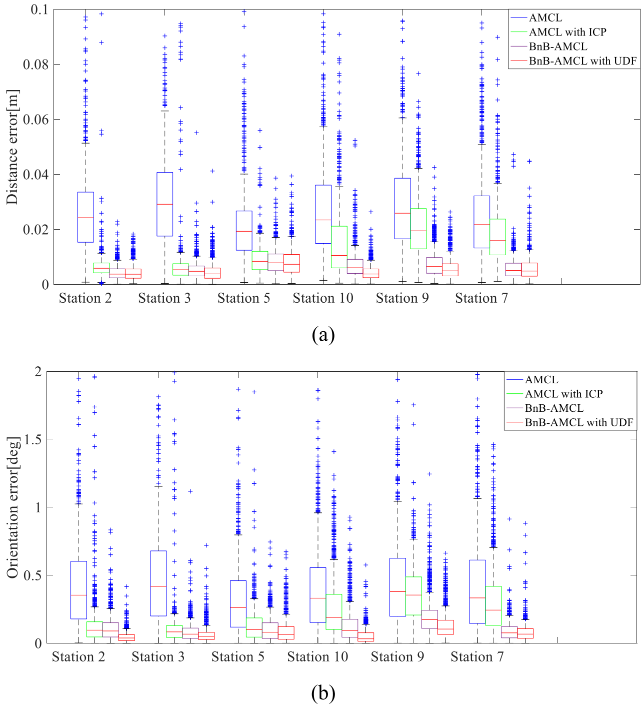


(a) (b)

FIGURE 14. QR code based visual positioning module: This module is used for obtaining the relative pose between the robot arm and the fixture with sub-millimeter accuracy. With this module, the positioning error of the mobile platform can be determined. In (a), the robot arm is capturing a QR code image which is shown in (b) as well as the positioning error.

TABLE 5. Localization accuracy using different methods.

Localization method	Average localization deviation		Standard deviation	
	Distance [m]	Orientation [deg]	Distance [m]	Orientation [deg]
AMCL	0.0288	0.4261	0.0258	0.3751
AMCL with ICP	0.0160	0.2471	0.0281	0.3937
BnB-AMCL	0.0066	0.1254	0.0053	0.1243
BnB-AMCL with UDF	0.0055	0.0917	0.0043	0.0940



(a)

(b)

FIGURE 15. Box plot of localization results using different localization methods: (a) Distance error. (b) Orientation error.

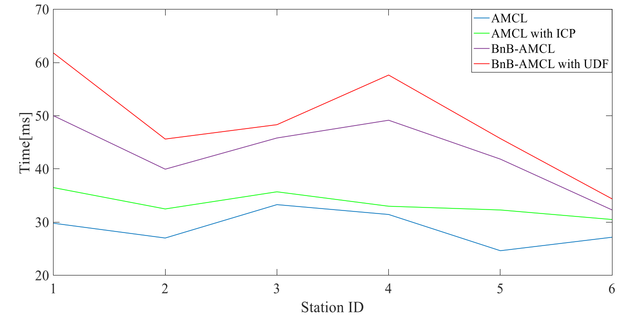


FIGURE 16. The average running time of different pose tracking methods.

use the same Lidar data set mentioned in global localization experiments. The localization deviation means the difference between a localization result and the average localization result among 500 times of localization. At each localization, a relative large odometer error following a uniform distribution $U(-0.1, 0.1)$ is introduced to simulate the odometer error during pose tracking. The bins size for AMCL is 0.1 m/5 deg while for BnB-AMCL is 0.8 m and the angular size is computed online according to (7). For positioning accuracy, the positioning error data from 463 loading and unloading tasks of the mobile manipulator in two weeks is adopted and all those tasks are finished successfully. One thing to note that the positioning error can be measured from the QR code based visual positioning module of our self-developed mobile manipulator, which is shown in Fig. 14. A more detailed description can be found in our previous work [36].

Table 5 shows an overview of localization accuracy and Fig. 15 shows a box plot that expresses the distribution of localization deviation of different methods. For clarity, the distance errors over 0.1m and orientation errors over 2 deg are ignored in our boxplot. Besides, all localization errors using our localization method are within such error range. As we can see from Table 5, BnB-AMCL with UDF has the best results regardless of average localization deviation and standard deviation. In addition, our BnB-AMCL is more accuracy than AMCL with ICP. From the box plot we can see that at some stations, the third quartile of AMCL with ICP is similar to the third quartile of BnB- AMCL, however, there are more outliers by using AMCL with ICP. This is because the ICP based scan match is susceptibility to the problem of local minima especially under the situation of large odometry error and multimodal target distribution. However, the above problems can be well solved by using the BnB-AMCL method.

The corresponding running time during the localization accuracy experiment is recorded, which are shown in Fig. 16. Since more steps are performed, our proposed pose tracking method inevitably requires more time than AMCL with ICP, where station 1 gets the maximum average runtime of 62 ms. However, 62 ms still meets the real-time requirement at the movement speed of 0.7m/s. Moreover, compared to AMCL with ICP, the localization performance in a harsh manufacture workshop is significantly improved through BnB-AMCL with UDF, hence it is totally worth it to spend a little more time.

To verify the impact of hybrid maps on pose tracking, localization results with hybrid maps and with single layer maps are compared. Using single layer maps means all maps

TABLE 6. Localization accuracy using different maps.

Map type	Average localization deviation		Standard deviation	
	Distance [m]	Orientation [deg]	Distance [m]	Orientation [deg]
Single layer maps	0.0067	0.1637	0.0050	0.2369
Hybrid maps	0.0055	0.0917	0.0043	0.0940

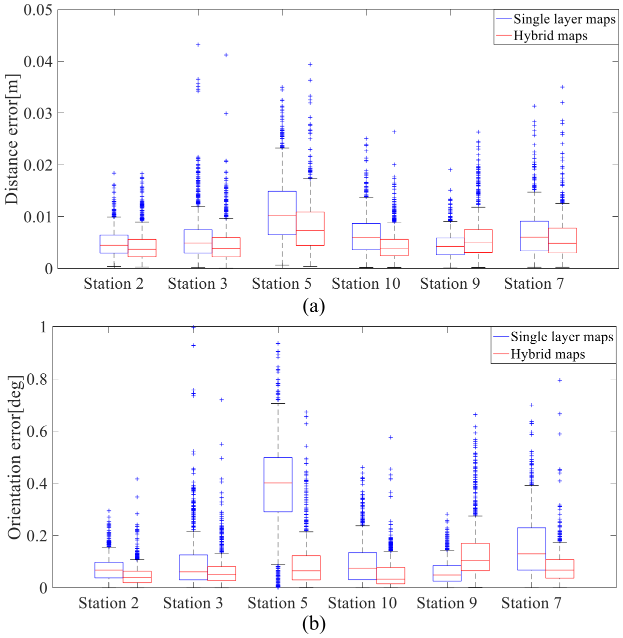


FIGURE 17. Box plot of localization results using different maps: (a) Distance error. (b) Orientation error.

for localization are created from an occupancy grid map which represents the lower plane of the two-layer structure of an environment. In this experiment, we only use BnB-AMCL with UDF as the localization method. Other localization methods with different types of maps will produce similar results. The results are shown in Table 6 and Fig. 17. As we can see, using hybrid grid maps can significantly reduce distance and orientation errors for most stations. However, the localization accuracy of using single-layer maps at station 9 is slightly better than that of using hybrid grid maps. This is because the ground near station 9 is flat and the Lidar data obtained at that station perfectly matches the single layer maps. But for a harsh manufacturing environment, the existence of flat ground is very accidental. Overall, our proposed method with hybrid maps has better performance for a randomly moving mobile manipulator in harsh manufacturing workshop. From the positioning accuracy experiments, the average positioning error of 0.005 m/0.111 deg and the standard deviation of 0.003 m/0.090 deg are obtained. The box plot of positioning error is shown in Fig. 18, where the maximum error is 0.016 m/0.528 deg and the third quartile of the distance error and the orientation error at all stations are below 0.01 m and 0.2 deg respectively. These results satisfy the accuracy of loading and unloading tasks. From the above experiments, it has been verified that our localization system can achieve real-time, reliable and high-precision pose tracing in harsh manufacturing workshops.

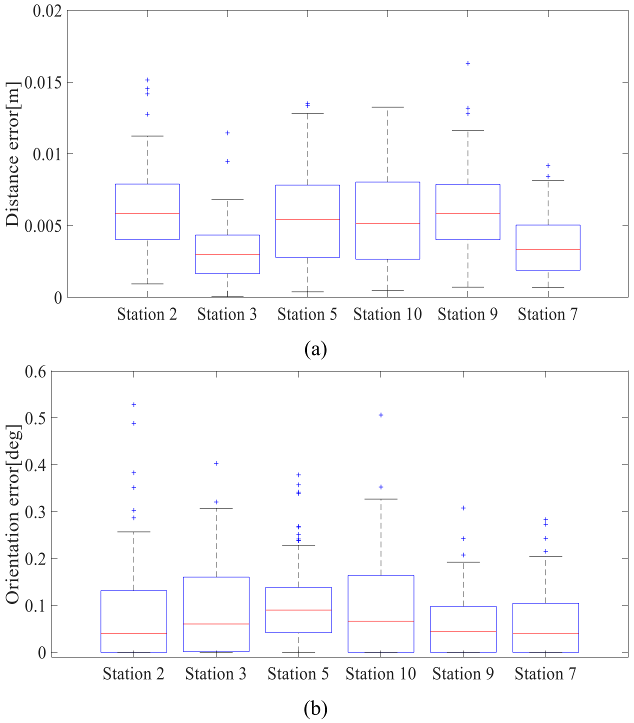


FIGURE 18. Box plot of positioning error at different stations: (a) Distance error. (b) Orientation error.

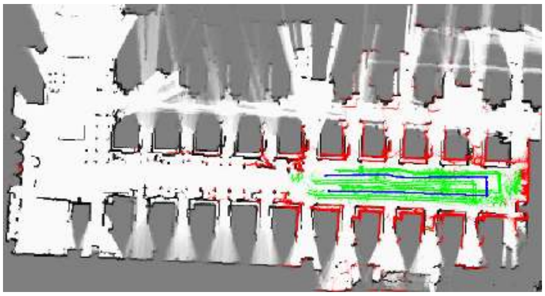


FIGURE 19. Localization results after robot moving along a trajectory: The blue curve shows the robot trajectory obtained by our localization system. The red point cloud shows the captured Lidar data during robot moving. All Lidar data is registered on the map according to our localization results. The green point cloud is the dynamic obstacle filtered by UDF. Note that part of green points forms a trajectory as well as our localization results. Those points are such measurements blocked by the mobile manipulator itself.

Fig. 19 shows the localization results of BnB-AMCL with UDF with all captured Lidar data after the mobile manipulator moving along a trajectory. All Lidar data is registered on the map according to the localization result. We can clearly see that randomly moving workers and other dynamic obstacles are effectively removed using UDF, as shown by the green point cloud. In addition, all registered Lidar data no matter caused by the higher or lower plane of the two-layer structure are all well-matched to the hybrid gird map.

VIII. CONCLUSION

In this paper, we present a hybrid maps enhanced localization system to address the self-localization problem of a mobile manipulator in harsh manufacturing workshops. The hybrid maps are used to model the two-layer structure of an environment and to improve the performance of localization. With such a localization system, the mobile manipulator can reliably load and unload workpieces for CNC in a harsh manufacturing workshop. The two main parts of our localization system are the CS based BnB-localization for global localization and the BnB-AMCL with the UDF method for pose tracking. Our global localization method adopts CS to sparse Lidar data and BnB search to achieve a fast global search. For an efficiency BnB search, the MLFs are employed to compute the match score and the upper bound. Moreover, we present that the computation method for the match score is the same computation method for the weight in AMCL and those bins generated by KLD-sampling can be used as the initial search space for BnB search. Thus, by combining BnB search and AMCL, we proposed the BnB-AMCL, which can achieve accuracy localization under the situation of large odometry error and multimodal target distribution. To future improve the pose tracking accuracy, two times of ICP based scan match are integrated into BnB-AMCL by using the hybrid point map. Finally, the unscented transform is adopted to improve the distance filter for mitigating the influence of dynamic obstacles such as moving workers. We evaluate our localization system through global localization experiments and pose tracking experiments in a real manufacturing workshop. The results demonstrate that our localization system has superior performance than many popular localization methods. Using the proposed localization system, our self-developed mobile manipulator had completed loading and unloading tasks in a manufacturing workshop for two weeks with a success rate of 100%.

REFERENCES

[1] I. Nielsen, Q.-V. Dang, G. Bocewicz, and Z. Banaszak, "A methodology for implementation of mobile robot in adaptive manufacturing environments," *J. Intell. Manuf.*, vol. 28, no. 5, pp. 1171–1188, Jun. 2017.

[2] X. Ding, Y. Liu, J. Hou, and Q. Ma, "Online dynamic tip-over avoidance for a wheeled mobile manipulator with an improved tip-over moment stability criterion," *IEEE Access*, vol. 7, pp. 67632–67645, 2019.

[3] G. Vasiljević, D. Miklić, I. Draganjac, Z. Kova ić, and P. Lista, "High-accuracy vehicle localization for autonomous warehousing," *Robot. Comput.-Integr. Manuf.*, vol. 42, pp. 1–16, Dec. 2016.

[4] C. Sprunk, B. Lau, P. Pfaff, and W. Burgard, "An accurate and efficient navigation system for omnidirectional robots in industrial environments," *Auton. Robots*, vol. 41, no. 2, pp. 473–493, Feb. 2017.

[5] H. Li and A. V. Savkin, "An algorithm for safe navigation of mobile robots by a sensor network in dynamic cluttered industrial environments," *Robot. Comput.-Integr. Manuf.*, vol. 54, pp. 65–82, Dec. 2018.

[6] X. Qu, B. Soheilian, and N. Paparoditis, "Vehicle localization using mono-camera and geo-referenced traffic signs," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Seoul, South Korea, Jun. 2015, pp. 605–610.

[7] M. Meilland, A. I. Comport, and P. Rives, "Dense omnidirectional RGB-D mapping of large-scale outdoor environments for real-time localization and autonomous navigation," *J. Field Robot.*, vol. 32, no. 4, pp. 474–503, Jun. 2015.

[8] J. M. Pak, C. K. Ahn, Y. S. Shmaliy, and M. T. Lim, "Improving reliability of particle filter-based localization in wireless sensor networks via hybrid particle/FIR filtering," *IEEE Trans. Ind. Informat.*, vol. 11, no. 5, pp. 1089–1098, Oct. 2015.

[9] F. Xiao, Z. Wang, N. Ye, R. Wang, and X.-Y. Li, "One more tag enables fine-grained RFID localization and tracking," *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, pp. 161–174, Feb. 2018.

[10] G. Li, J. Meng, Y. Xie, X. Zhang, L. Jiang, and Y. Huang, "An improved observation model for Monte-Carlo localization integrated with reliable reflector prediction," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Hong Kong, Jul. 2019, pp. 972–977.

[11] P. Egger, P. V. K. Borges, G. Catt, A. Pfrunder, R. Siegwart, and R. Dube, "PoseMap: Lifelong, multi-environment 3D LiDAR localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Madrid, Spain, Oct. 2018, pp. 3430–3437.

[12] Y. Xu, Y. S. Shmaliy, Y. Li, X. Chen, and H. Guo, "Indoor INS/LiDAR-based robot localization with improved robustness using cascaded FIR filter," *IEEE Access*, vol. 7, pp. 34189–34197, 2019.

[13] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust Monte Carlo localization for mobile robots," *Artif. Intell.*, vol. 128, nos. 1–2, pp. 99–141, May 2001.

[14] J.-L. Blanco, J. González-Jiménez, and J.-A. Fernández-Madrigal, "A robust, multi-hypothesis approach to matching occupancy grid maps," *Robotica*, vol. 31, no. 5, pp. 687–701, Aug. 2013.

[15] S. Park and K. S. Roh, "Coarse-to-fine localization for a mobile robot based on place learning with a 2-D range scan," *IEEE Trans. Robot.*, vol. 32, no. 3, pp. 528–544, Jun. 2016.

[16] H. Liu, T. Liu, Y. Li, M. Xi, T. Li, and Y. Wang, "Point cloud registration based on MCMC-SA ICP algorithm," *IEEE Access*, vol. 7, pp. 73637–73648, 2019.

[17] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Stockholm, Sweden, May 2016, pp. 1271–1278.

[18] R. W. Wolcott and R. M. Eustice, "Robust LIDAR localization using multiresolution Gaussian mixture maps for autonomous driving," *Int. J. Robot. Res.*, vol. 36, no. 3, pp. 292–319, Mar. 2017.

[19] J. Saarinen, J. Paanajarvi, and P. Forsman, "Best-first branch and bound search method for map based localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Francisco, CA, USA, Sep. 2011, pp. 59–64.

[20] D. Fox, "Adapting the sample size in particle filters through KLD-sampling," *Int. J. Robot. Res.*, vol. 22, no. 12, pp. 985–1003, Dec. 2003.

[21] R. P. Guan, B. Ristic, L. Wang, and J. L. Palmer, "KLD sampling with Gmapping proposal for Monte Carlo localization of mobile robots," *Inf. Fusion*, vol. 49, pp. 79–88, Sep. 2019.

[22] C. Wu, T. A. Huang, M. Muffert, T. Schwarz, and J. Grater, "Precise pose graph localization with sparse point and lane features," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Vancouver, BC, Canada, Sep. 2017, pp. 4077–4082.

[23] J. Biswas, B. Coltin, and M. Veloso, "Corrective gradient refinement for mobile robot localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Francisco, CA, USA, Sep. 2011, pp. 73–78.

[24] J. Rowekamper, C. Sprunk, G. D. Tipaldi, C. Stachniss, P. Pfaff, and W. Burgard, "On the position accuracy of mobile robot localization based on particle filters combined with scan matching," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Vilamoura, Portugal, Oct. 2012, pp. 3158–3164.

[25] S. Thrun, "Learning occupancy grid maps with forward sensor models," *Auton. Robots*, vol. 15, no. 2, pp. 111–127, Sep. 2003.

[26] G. D. Tipaldi, D. Meyer-Delius, and W. Burgard, "Lifelong localization in changing environments," *Int. J. Robot. Res.*, vol. 32, no. 14, pp. 1662–1678, Dec. 2013.

[27] G. Li, J. Meng, Y. Xie, X. Zhang, Y. Huang, L. Jiang, and C. Liu, "Reliable and fast localization in ambiguous environments using ambiguity grid map," *Sensors*, vol. 19, no. 15, p. 3331, Jul. 2019.

[28] P. Merriaux, Y. Dupuis, R. Boutteau, P. Vasseur, and X. Savatier, "Robust robot localization in a complex oil and gas industrial environment," *J. Field Robot.*, vol. 35, no. 2, pp. 213–230, Mar. 2018.

[29] G. Vasiljevic, F. Petric, and Z. Kovacic, "Multi-layer mapping-based autonomous forklift localization in an industrial environment," in *Proc. 22nd Medit. Conf. Control Automat.*, Palermo, Italy, Jun. 2014, pp. 1134–1139.

[30] M. A. Jayaram and H. Fleyeh, "Convex hulls in image processing: A scoping review," *Amer. J. Intell. Syst.*, vol. 6, no. 2, pp. 48–58, 2016.

[31] P. Pfaff, C. Plagemann, and W. Burgard, "Gaussian mixture models for probabilistic localization," in *Proc. IEEE Int. Conf. Robot. Automat.*, Pasadena, CA, USA, May 2008, pp. 467–472.

[32] A. I. Eliazar and R. Parr, "Learning probabilistic motion models for mobile robots," in *Proc. 21st Int. Conf. Mach. Learn. (ICML)*, Banff, AB, Canada, 2004, p. 32.

[33] Z. Zhang, X. Zhang, H. Pan, W. Salman, Y. Rasim, X. Liu, C. Wang, Y. Yang, and X. Li, "A novel steering system for a space-saving 4WS4WD electric vehicle: Design, modeling, and road tests," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 1, pp. 114–127, Jan. 2017.

[34] J. Biswas and M. M. Veloso, "Episodic non-Markov localization," *Robot. Auton. Syst.*, vol. 87, pp. 162–176, Jan. 2017.

[35] D. Sun, F. Geisser, and B. Nebel, "Towards effective localization in dynamic environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Daejeon, South Korea, Oct. 2016, pp. 4517–4523.

[36] J. Meng, S. Wang, G. Li, L. Jiang, X. Zhang, and Y. Xie, "A convenient pose measurement method of mobile robot using scan matching and eye-in-hand vision system," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Hong Kong, Jul. 2019, pp. 1062–1067.

GEN LI received the B.S. degree in mechanical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2014, where he is currently pursuing the Ph.D. degree with the School of Mechanical Science and Engineering.

He has published ten academic journal and conference papers. His research interests include robot perception, information fusion, and robot programming.

YU HUANG received the B.S., M.S., and Ph.D. degrees from the School of Mechanical Science and Engineering, Huazhong University of Science and Technology (HUST), Wuhan, China, in 1993, 2000, and 2004, respectively.

He is currently a Professor with the School of Mechanical Science and Engineering, HUST, where he is also an Assistant Director of the National Engineering Research Center of Manufacturing Equipment Digitization. He has published more than 80 articles, and hold more than 30 patents. His research interests include mechatronics, mobile robot, mechanical design, and numerical control technology.

XIAOLONG ZHANG received the master's degree in mechanical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2015, where he is currently pursuing the Ph.D. degree with the School of Mechanical Science and Engineering.

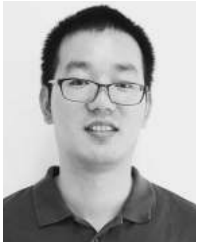
His current research interests include mobile robot, robot control, and motion planning.

CHAO LIU received the B.S. degree in mechanical engineering from Northeastern University, Shenyang, China, in 2013, and the M.S degree in mechatronic engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2018, where he is currently pursuing the Ph.D. degree with the School of Mechanical Science and Engineering.

His research interests include computer vision and mobile robot navigation.

WENJUN SHAO received the B.S. degree in mechanical engineering and automation from the Wuhan University of Science and Technology, Wuhan, China, in 2012, and the Ph.D. degree in mechanical and electronic engineering from the Huazhong University of Science and Technology, Wuhan, in 2016.

He is currently a Postdoctoral Researcher with the School of Mechanical Science and Engineering, Huazhong University of Science and Technology. His research interests include multiaxis motion control, automatic instrumentation, image sensor, and image processing.



LIQUAN JIANG received the B.S. degree in mechanical engineering from the China University of Geosciences, Wuhan, China, in 2014, and the M.S. degree in mechanical and electronic engineering from the Huazhong University of Science and Technology, Wuhan, in 2017, where he is currently pursuing the Ph.D. degree with the School of Mechanical Science and Engineering.

His research interests include optimization algorithms and path planning.

JIE MENG received the B.S. degree in mechanical engineering from the Wuhan University of Technology, Wuhan, China, in 2016. He is currently pursuing the Ph.D. degree with the School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan.

He has published six academic journal and conference papers, and holds eight patents. His research interests include robot perception and mobile robot navigation.