

A high-efficiency, information-based exploration path planning method for active simultaneous localization and mapping

Peng Li^{1,2} , Cai-yun Yang¹, Rui Wang¹ and Shuo Wang^{1,2,3}

Abstract

The efficiency of exploration in an unknown scene and full coverage of the scene are essential for a robot to complete simultaneous localization and mapping actively. However, it is challenging for a robot to explore an unknown environment with high efficiency and full coverage autonomously. In this article, we propose a novel exploration path planning method based on information entropy. An information entropy map is first constructed, and its boundary features are extracted. Then a Dijkstra-based algorithm is applied to generate candidate exploration paths based on the boundary features. The dead-reckoning algorithm is used to predict the uncertainty of the robot's pose along each candidate path. The exploration path is selected based on exploration efficiency and/or high coverage. Simulations and experiments are conducted to evaluate the proposed method's effectiveness. The results demonstrated that the proposed method achieved not only higher exploration efficiency but also a larger coverage area.

Keywords

Active SLAM, exploration, information-based, simultaneous localization and mapping, path planning

Date received: 9 July 2019; accepted: 26 December 2019

Topic: Mobile Robots and Multi-Robot Systems

Topic Editor: Yangquan Chen

Associate Editor: Hongbo Gao

Introduction

In recent years, autonomous exploration path planning and simultaneous localization and mapping (SLAM) in an unknown environment has drawn increasing attention from robotics and computer vision communities.^{1–6} In the research fields of driverless cars,^{7–9} unmanned aerial vehicles, bionic robot,¹⁰ and household robot, a real-time, high-precision environmental map is the basis for these applications to achieve other high-level jobs such as navigation, surveying, and picking and dropping items. However, the actual application scenarios are ever-changing, and it is impossible to preestablish maps for each scene. The lack of autonomous environmental exploration capabilities will significantly limit the widespread use of robot technology. Therefore, many scholars began to study the

robot's autonomous exploration path planning problem in an unknown environment, enabling the robot to explore and establish the map actively.^{11–21} They combine the local observation data with a partially established map to get

¹ State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China

² University of Chinese Academy of Sciences, Beijing, China

³ Center for Excellent in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Beijing, China

Corresponding author:

Shuo Wang, State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Zhongguancun East Road No. 95, Haidian District, Beijing 100190, China.
Email: shuo.wang@ia.ac.cn



an optimal exploration trajectory. The robot autonomously conducts mapping and navigation along this trajectory. Some scholars have also considered the optimization of the sensor's pose during the exploration.^{22–24} The classic frontier-based approach^{11,14,25} can achieve a full coverage observation of the environment by selecting the nearest accessible and unvisited frontier, but it is a one-step greedy method and cannot guarantee that the final exploration trajectory is optimal. The exploration efficiency may decrease significantly due to frequent retrace the route and the fragmentation of the unexplored region. The possibility of positioning loss will also increase without constraints of the uncertainty in the robot's location. Some scholars have adopted an optimization-based multistep approach to obtain the optimal path by maximizing the information gain during the exploration.^{15,23,26–30} However, the computation cost increases significantly as the length of the generated path increases. As a result, the path length has to be limited, or the map resolution is reduced. The optimization-based methods tend to have a high efficiency of exploration at the beginning, but at the end of the exploration, as the fragmentation of the exploration environment increases, the exploration efficiency reduces drastically.

In this article, a novel exploration path planning method based on information entropy has been proposed. The proposed method tries to follow the boundary of the unexplored space and the explored space, avoid breaking the unexplored region into pieces and maintain the integrity of the unexplored area. This mechanism achieves a higher and more stable exploration efficiency throughout the exploration process. Information entropy can fully reflect the quantity of information, which is more accurate and has more advantages than the semantic concepts in the frontier-based methods. First, this representation for the environment makes the proposed method not only has the same global convergence as the typical frontier-based methods but also overcomes the disadvantage of the efficiency reduction due to the fragmentation. Second, the representation simplifies the calculation of information gain. The information entropy gain can be obtained by a more straightforward summation operation rather than a large number of multiplication operations as needed to calculate the posterior probability. A Dijkstra algorithm-based exploration path generation strategy is proposed to get a set of candidate paths under the consideration of the robot's pose uncertainty along these paths. Then an efficiency-optimal selection strategy is used to choose the final exploration path. The results of simulations and experiments show that the proposed method has a significant improvement compared to two classic exploration methods.

The main contributions of this article are summarized as follows:

- (1) A path generation method based on information entropy maps and boundary features is proposed to generate efficient exploration paths.
- (2) Uncertainty changes in the pose of the robot are also taken into account in the candidate exploration paths.
- (3) Practical validation of the proposed method on a mobile robot.

Related work

The exploration in an unknown environment to active SLAM have been studied by many researchers.^{11,14–17,25,26,29,31–42} We will give a brief review of these methods as the following three categories.

Frontier-based exploration methods

The frontier-based exploration methods^{11,14,25,38,42} in unknown scenes can be traced back to the seminal work of Yamauchi.¹¹ The frontier refers to the boundary between the open space and unexplored space. The robot explores new areas by moving to the nearest frontier until all the frontier in the environment has been explored. They also extend the frontier-based methods to multiple robots.⁴² Freda and Oriolo mainly use a data structure that they called the Sensor-based Random Tree (*SRT*) to create the exploration path and uses the frontier to guide the growth of the *SRT* toward unexplored areas.¹⁴ Mobarhani et al. propose a histogram-based evaluation method to cluster the frontier cell and select the optimal target. They evaluate each cluster by calculating the distance from each cluster center to the robot, and the number of cells included in each cluster.²⁵ By using the frontier-based method, the order of observations is not constrained, which results in the area to be explored that may be gradually divided into more and more small areas. In order to achieve full coverage of the environment, the robot has to jump back and forth between these small areas. However, the classic frontier-based method tends to cause an increase in the fragmentation of the unexplored region. We propose a method of exploration path planning based on boundary features. The proposed method can avoid dividing the region into multiple small pieces and reducing the fragmentation.

Environmental characteristics-based method

Mu et al. use the features extracted from the environment to construct the geometric representation of the environment, which they called topological feature graph (TFG).⁴³ They use a sampling-based method to generate the exploration path with TFG. Xu et al. use the time-varying tensor field to represent the environment and guide the robot to move.²³ At the same time, an RGB-D camera is attached to the robot arm to scan the environment, and the camera's 3-D trajectory is optimized simultaneously to move smoothly to

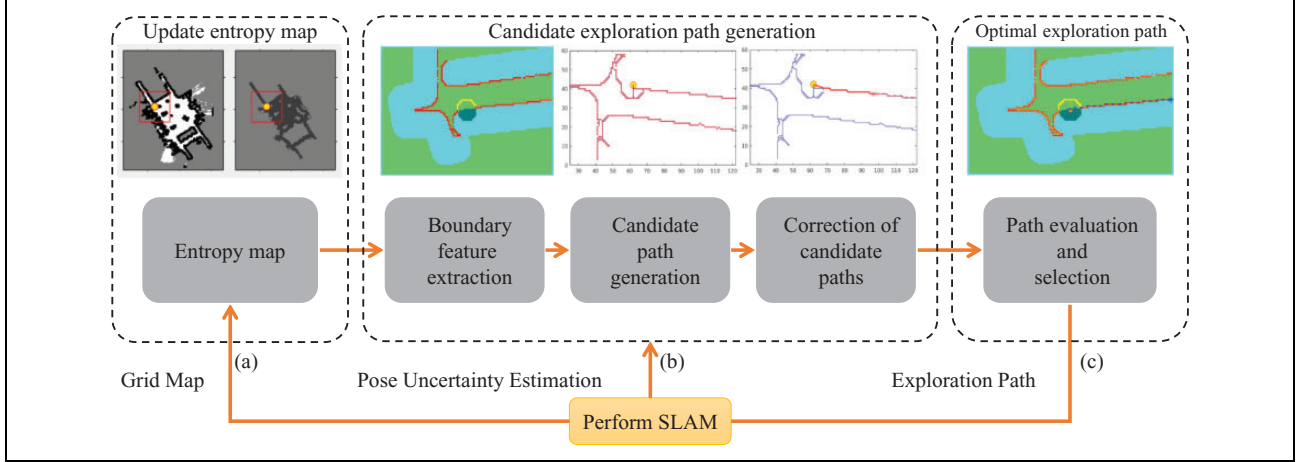


Figure 1. Overall scheme of our exploration path planning approach for the active SLAM. In Figure (a), the local grid map and sensor data are used to update the information entropy map incrementally. In Figure (b), the boundary features are extracted, and a set of candidate exploration paths is generated. The length and number of these candidate paths are corrected. In Figure (c), these candidate paths are evaluated by a comprehensive path selection strategy, and an optimal exploration path is obtained.

obtain high-quality scans. Vallvé et al. build potential information filed by evaluating the path and map entropy reduction. Then they compute an exploration path in the gradient descent direction of the potential information filed.²⁸ We extract the boundary features of the environment to generate the exploration path. The boundary feature is stable and has lower computational complexity than the method¹¹ and the method.²⁹

Information-theoretic methods

Leung et al. treat the robot's exploration path planning in an unknown environment as an optimal path planning.²⁶ They assume that the features in the environment are static, and use model predictive control to obtain a control sequence of a robot by maximizing the information entropy gain. Vallvé et al. propose a method based on rapidly exploring random tree (RRT).⁴⁰ They use RRT to generate a set of candidate paths, and then they use POSE SLAM to evaluate these candidate paths by predicting changes in the uncertainty of the robot's path and the information entropy of the map to select the final exploration path. Stachniss et al. use a particle filter to predict changes in maps and robot's poses after moved to each possible position.³⁷ They proposed a method to select the exploration actions by trading off between the expected information gain and the possible sensor observation. Bourgault et al. propose a weighted utility evaluation function based on the uncertainty of the localization and information gain, and used the optimization method to obtain the path of exploration.¹⁵ As the length of the planning path increases, the computational cost will increase dramatically. Bai et al. propose a Bayesian optimization-based approach.²⁹ Firstly, some random exploration targets around the robot are generated, and mutual information at these targets are calculated.

Moreover, a Gaussian process (GP) is trained to predict the distribution of mutual information. After obtaining the initial GP model, they use the Bayesian optimization and the GP model to find the best exploration target, and the mutual information moving to it and add them to the training data set to train a new GP model. Then they use this model to calculate the best target and mutual information. This iterative process is repeated several times. Finally, the final exploration goal is obtained. Since the acquisition of the posterior probability is a computationally complex calculation, the existing methods mostly address this problem by using a shorter planning path or reducing the resolution of the map. However, the boundary features used in this article cover only a small portion of the map. Therefore, the number of planning paths is significantly reduced.

Approach overview

We refer to the problem that a robot plans exploration paths autonomously in an unknown environment and performs simultaneous mapping and localization as the active SLAM. The active SLAM can be separated into two subproblems, SLAM and the active exploration path planning. The traditional SLAM approach passively receives observation data and solves the problem of localization and mapping. Based on the partial information of the existing environment, the active algorithm plans an efficient path for the next exploration steps. This path must be traded off between the benefits of exploring new areas and revisiting areas to improve accuracy. In order to ensure that the exploration process does not fail, the uncertainty of the robot's pose will also be added to the constraints of the exploration path planning.

Figure 1 shows the pipeline of our proposed method. In this article, an open-source SLAM⁴⁴ has been used to provide an occupied grid map, robot positioning, and the

uncertainty estimation of the robot's pose. In Figure 1(a), the local grid map and sensor data are used to update the information entropy map incrementally. Changes in the grid map will be updated synchronously to the information entropy map. The steps for generating the candidate paths are as shown in Figure 1(b). First, the boundary features in the grid map and the information entropy map will be extracted by a convolution operation and a threshold segmentation algorithm. Then, a path generation algorithm based on the extracted boundary features will be used to generate a set of candidate exploration paths. The length and number of these candidate paths are modified by the candidate path correction module to produce a set of candidate exploration paths that cover the unexplored area. In Figure 1(c), the candidate paths obtained from Figure 1(b) are evaluated by the proposed comprehensive path selection strategy, and finally, an optimal exploration path is obtained. We repeat the above process until the information entropy of all the grids in the information entropy map is reduced to a sufficiently low threshold.

Information entropy map

We construct a grid information entropy map from the occupied grid map and the sensor data. The entropy map is used to represent the perception of the environment. The occupied grid map can only give a state of each grid its occupied or idle state, and it is susceptible to noise as the state of the grid changes too fast with the observed data, which has a significant impact on the path planning processing. A continuous information entropy map is created and updated synchronously with the SLAM process to solve the issue. The entropy of a grid is gradually and smoothly updated as the observation progresses, which reflects the average state of the observations, and thus, the grid information entropy map is less sensitive to noise.

Information entropy is a quantitative measure of information. For a discrete random variable X with possible values x_1, x_2, \dots, x_n , and probability mass function, the information entropy of the random variable X is defined as

$$\hat{H}(X) = - \sum_{i=1}^n P(x_i) \log(P(x_i)) \quad (1)$$

For each grid, its information entropy is calculated and continuously updated as the observation progresses. There are only two states in each grid, that is, idle or occupied, so the information entropy in this article is defined as

$$\hat{H}(c_i) = -P(c_i) \ln P(c_i) - (1 - P(c_i)) \ln(1 - P(c_i)) \quad (2)$$

where $P(c_i)$ refers to the probability that the grid cell c_i is occupied. The information entropy map is continuously updated with the smoothed entropy value using equation (3)

$$H_{t+1}(c_i) = k H_t(c_i) + (1 - k) \hat{H}_t(c_i) \quad (3)$$

where $H_t(c_i)$ is the entropy of the grid cell c_i in the grid information entropy map at time step t and $\hat{H}_t(c_i)$ is the estimated entropy value for cell c_i , $H_{t+1}(c_i)$ is the updated entropy for the grid cell c_i . k is an updated coefficient that determines the update weights of the actual entropy and the observations.

Candidate exploration path generation

The active SLAM is mainly concerned with how to use local observation information to plan the exploration path, and simultaneously locate itself and establish a complete environment map with or without a prior environmental map. In this section, we will first introduce the candidate path generation method based on the local observation information, and then we will give the final optimal exploration selection strategy in the next section.

Boundary feature extraction

After obtaining data from the sensor, we update the probability estimations in the occupied grid map \mathbf{I}_g and the related information entropy map \mathbf{I}_e . The maps are shown in Figure 2(a) and (b), respectively.

We obtain candidate exploration paths based on two kinds of boundary features. The entropy map boundary feature \mathbf{B}_e in this article refers to the grids in the high entropy region are adjacent to the low entropy region. The entropy grid map boundary feature \mathbf{B}_g is the grids in the free area and adjacent to the occupied grids in the grid map. It should be noted that the grid map here is obtained from GMapping⁴⁴ and has been inflated. The boundary features can be extracted from the information entropy map \mathbf{I}_e or the inflated grid map \mathbf{I}_g , respectively. The boundary feature is shown in Figure 2(c). It is a bit like *frontier* but different. Firstly, the source of boundary features is different. The boundary features come from the information entropy map and the occupied grid map. Secondly, the use of features is different. The boundary features are not used to find an exploring goal, but to plan the candidate path for exploration. Unlike frontier-based methods, our approach can plan the path back to the previous area to limit the uncertainty of the robot's pose.

We first select an appropriate partition threshold to divide the information entropy map into high information entropy areas and low information entropy areas. A con-

volution kernel $\mathbf{M} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}_{3 \times 3}$ will be used to con-

volve the map \mathbf{I}_e and get the convolution map \mathbf{I}_{ec} . Similarly, we can also get the map \mathbf{I}_{gc} from the grid map convolution. Then the boundary feature \mathbf{B}_e and \mathbf{B}_g are extracted through the following formula

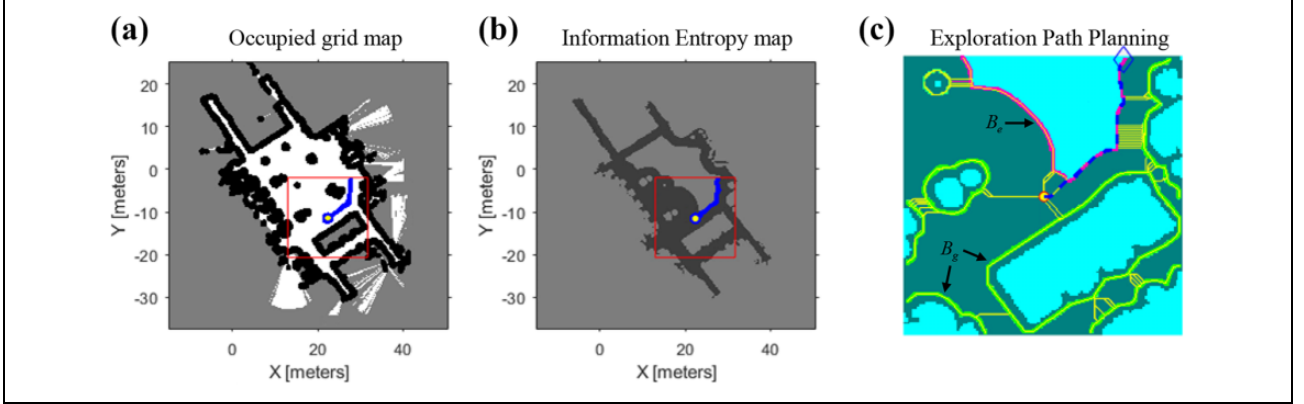


Figure 2. Boundary feature extraction and exploration path determination. We build and update an information entropy map shown in Figure (b) with the same scale as the occupied map shown in Figure (a). Figure (c) presents the area marked by the red box in Figure (a). In Figure (c), the thick pink line B_e describes the boundary feature extracted from the information map Figure (b), and the thick green line B_g presents the boundary feature extracted from the occupied grid map Figure (a). The thin yellow lines show the generated candidate paths.

$$B_e = \{X | I_{ec}(X) < n^2, I_e(X) > 0, I_g(X) \neq 1\}$$

$$B_g = \{X | I_{gc}(X) < n^2, I_g(X) \neq 1\} \quad (4)$$

where n is the dimension of the convolution kernel M , X is the coordinate on the grid.

The frontier-based method moves toward to the frontier to explore, and the proposed method is to try to explore along with the proposed boundary feature. Different from the classic *frontier* concept, we have a boundary feature that is not available in frontier-based methods, that is, the boundary feature extracted from the grid map B_g . These two kinds of boundary features allow us to not only explore the unknown region but also return to the previous area to relocate or improve the quality of the existed map.

Different direction selection strategies result in significant efficiency differences. The proposed method tends to plan a path that tries to surround the unexplored area, rather than directly go toward the unexplored areas and fragment it. Hence, we have fewer repetitive observations to improve coverage and keep a higher exploration efficiency. Simultaneously, we consider the optimal distance of observation and the robot's positioning uncertainty in path planning.

dpi is an abbreviation for dots per inch, which is a measure of sensor scanner dot density. For multiline laser sensors, the scanned laser beams are not necessarily parallel but always arranged at a certain angle θ .

We define an analogous parameter S_{dpi} to reflect the scanner density in equation (5)

$$S_{dpi} = (D \cdot \tan \theta)^{-1} \quad (5)$$

where θ is the angle between the two scan beams. D is the distance between the object and the laser sensor. It is shown in Figure 3. It should be declared that the observation of the sensor is not all effective. Here the effective observation radius R_e is defined as

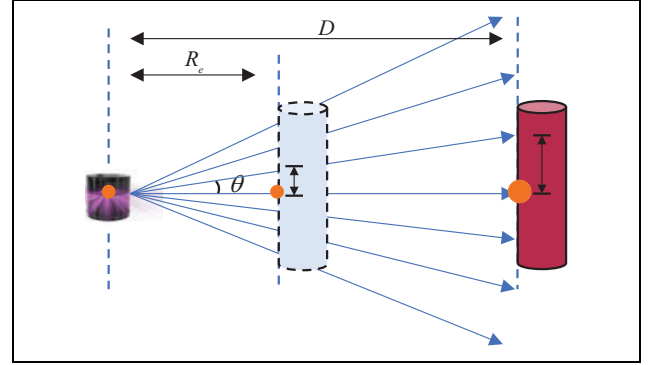


Figure 3. The definition of the effective radius R_e of sensor detection. θ is the angle between two adjacent scan lines.

$$\{R_e | \text{dist}(x, x_0) < R_e, S_{dpi}(x) > dpi_{\min}, x, x_0 \in \mathbb{R}^2\} \quad (6)$$

where x is the sensor point in the robot base coordinate system, x_0 is the origin of the coordinate. The closer to the laser sensor, the scanning points are denser, and the farther they will be sparser.

Only about half of the area is unexplored when exploring along the boundary, which is shown in Figure 4. The overlapping areas were observed almost twice. It is generally not optimal to use boundary features for candidate path generation. Thus we introduced a parameter α to handle the overlap problem. We use an extern radius $R_u = \alpha R_e$ to update the information entropy map I_e . We found that there is a close relationship between the updated radius of R_u and the overlap. The coefficient α can directly adjust the overlap, which in turn affects the efficiency of the exploration.

When the updated radius is twice the effective observations radius, that is, $R_u = 2R_e$, half of the updated information entropy map overlaps. The area marked in red in Figure 4 shows the overlapping area between adjacent

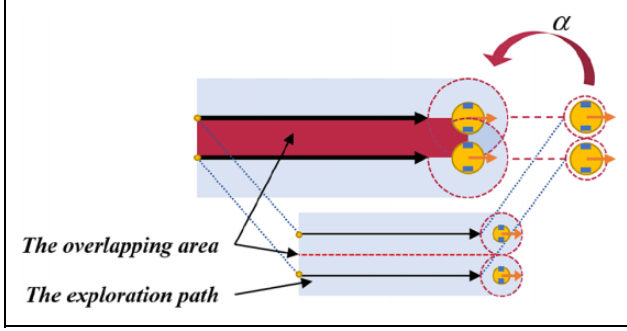


Figure 4. The overlapping area between two adjacent observations. By adjusting the coefficient α , we can control the overlap. When α is set to 2.0, the overlapping areas disappear.

observations. In the lower part of Figure 4, we can see that the overlap between adjacent observations has disappeared. If we adjust the coefficient α to make that $R_u = \alpha R_e$, then the overlap between adjacent observation areas also changes, and the exploration efficiency is correspondingly affected. The relationship between overlapping area ratio R_{ov} and parameter α is defined as equation (7)

$$R_{ov} = \begin{cases} 1 - 0.5\alpha, & 0 < \alpha < 2.0 \\ 0, & \alpha \geq 2.0 \end{cases} \quad (7)$$

When we increase the value of α , the overlapping area ratio R_{ov} will decrease, but the exploration efficiency will increase, and vice versa. Therefore, by selecting the coefficient α , we can balance the efficiency and quality of the exploration.

Candidate path generation

As mentioned above, the exploration along the boundaries of information entropy is very efficient in an unknown environment. We use the *Dijkstra* algorithm and the constructed graph G to generate paths along the boundaries.

In this article, we use a graph structure to represent the environment, which is similar to J Vallé's.⁴⁰ A graph $G(N, E, W)$ consists of edges and nodes, where N represents nodes, each of which is a grid cell in the grid map, E is the edge between nodes, and W refers to the edge weight.

We treat the current position of the robot on the grid map as the starting node and the grids on the boundary as the ending nodes. Then we use the *Dijkstra* algorithm to obtain candidate paths. In order to make planned paths along the boundary as much as possible and to actively avoid the collision, we increase the transition cost to nonboundary points and simultaneously reduce the cost to the boundary points in the graph G . This modification makes the path planning algorithm more inclined to select the path along the extracted boundary and make the generated path inherently avoid an obstacle. We adjust the weight of the edge between the node n_i and its adjacent node n_j

$$w_{ij} = \begin{cases} k_1 \cdot w_{ij}, & n_j \in B_e \\ k_2 \cdot w_{ij}, & n_j \in B_g \end{cases} \quad (8)$$

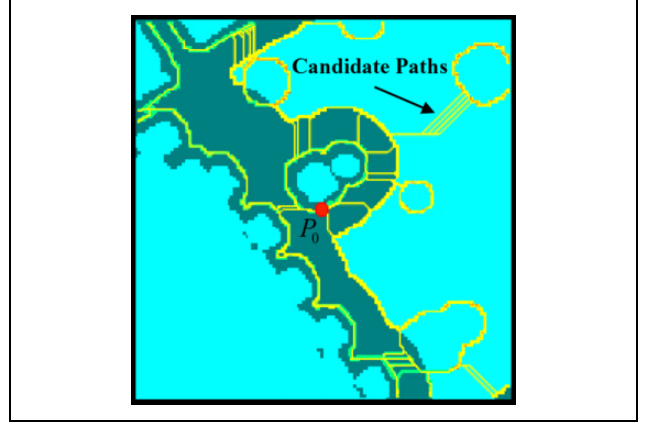


Figure 5. Candidate paths generation. The red marker point P_0 shows the current position of the robot. The yellow lines in the figure are the candidate paths.

where $n_j \in Neighbor(n_i)$ and $n_i, n_j \in B_e \cup B_g$, $0 \leq k_1 < k_2 < 1.0$.

The *Dijkstra* algorithm is very efficient and suitable for candidate path generation from the extracted boundaries because it only needs to be run once to get all the candidate paths. A case in point P_0 is shown in Figure 5. The thin yellow lines are the original candidate paths that grow along the boundaries as much as possible and have the obstacle avoiding ability. They can pass through the free space as need. So we can see that there are some connections between the different boundaries, which are shown as the yellow lines.

Correction of candidate paths based on uncertainty estimation

Rodriguez-Arevalo et al.⁴⁵ reported that the monotonicity of spatial propagation of uncertainty is preserved when the determinant of the covariance matrix is used as criteria in 2-D space. Inspired by their work, we use the dead-reckoning algorithm to estimate the covariance matrix and use the determinant of the covariance matrix as a criterion to quantify the uncertainty of the robot's pose as the robot moves. We constrain the uncertainty of the candidate path by choosing a threshold according to the monotonicity property.

The pose of the coordinate j relative to the coordinate system i can generally assume to be a Gaussian distribution. The estimated pose is a vector $\hat{X}_{ij} = (x_{ij}, y_{ij}, \phi_{ij})$ and the associated covariance matrix is Σ_{ij} . $X_{jk} = (x_{jk}, y_{jk}, \phi_{jk})$ is another pose which is respect to coordinate j . The calculation of X_{ik} from the state X_{ij} and X_{jk} is defined as

$$X_{ik} = X_{ij} \oplus X_{jk} \quad (9)$$

The covariance matrix Σ_{ik} can be approximated by

$$\Sigma_{ik} \approx J_{\oplus} \begin{bmatrix} \Sigma_{ij} & 0 \\ 0 & \Sigma_{jk} \end{bmatrix} J_{\oplus}^T \quad (10)$$

where the *Jacobian* of the compounding operation in equation (10) is

$$J_{\oplus} = \frac{\partial X_{ik}}{\partial (X_{ij}, X_{jk})} = \begin{bmatrix} 1 & 0 & y_{ij} - y_{ik} & \cos(\phi_{ij}) & -\sin(\phi_{ij}) & 0 \\ 0 & 1 & x_{ik} - x_{ij} & \sin(\phi_{ij}) & \cos(\phi_{ij}) & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \quad (11)$$

Let $J_{\oplus} = [J_{1\oplus}, J_{2\oplus}]$, and $J_{1\oplus}, J_{2\oplus}$ are given by

$$J_{1\oplus} = \begin{bmatrix} 1 & 0 & y_{ij} - y_{ik} \\ 0 & 1 & x_{ik} - x_{ij} \\ 0 & 0 & 1 \end{bmatrix} \quad (12)$$

$$J_{2\oplus} = \begin{bmatrix} \cos(\phi_{ij}) & -\sin(\phi_{ij}) & 0 \\ \sin(\phi_{ij}) & \cos(\phi_{ij}) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

So from equation (10) we can get

$$\Sigma_{ik} \simeq J_{1\oplus} \Sigma_{ij} J_{1\oplus}^T + J_{2\oplus} \Sigma_{jk} J_{2\oplus}^T \quad (13)$$

$$\det(J_{1\oplus}) \equiv 1, \det(J_{2\oplus}) \equiv 1 \quad (14)$$

Carrillo et al.¹⁷ have reported that the monotonicity of the uncertain criteria is preserved under linearized assumptions when using the determinant of the covariance matrix as the uncertain criteria and equation (14) is established.

When the pose of the robot changes from step $n-1$ to the next step n , the relative motion can be compounded with the relative pose ${}^{(n-1)}\xi_n \sim (\delta_x, \delta_y, \delta_\phi)^T$. $\delta_x, \delta_y, \delta_\phi$ is the change in distance and steering from step $n-1$ to n . The position of the robot concerning the odometry frame is

$${}^{(0)}\xi_n = {}^{(0)}\xi_1 \oplus {}^{(1)}\xi_2 \oplus \dots \oplus {}^{(n-1)}\xi_n \quad (15)$$

We assume that the distribution of ${}^{(n-1)}\xi_n$ is a zero-mean Gaussian with the covariance matrix V_n and the associated variance for every step is defined as

$$V_n = \begin{bmatrix} ({}^{(n)}\delta_d)^2 & 0 & 0 \\ 0 & ({}^{(n)}\delta_d)^2 & 0 \\ 0 & 0 & ({}^{(n)}\delta_\phi)^2 \end{bmatrix} \quad (16)$$

where ${}^{(n)}\delta_d$ is the distance traveled at step n . The covariance matrix V_n reflects that the uncertainty increases as the distance increases and the angle changed in heading increases. We use GMapping to build a grid map and provide the robot's location. The initial covariance matrix Σ_0 is obtained by the current particle distribution in the GMapping. The current covariance matrix can be updated by the last estimation Σ_{n-1} and the new V_n

$$\Sigma_n \simeq J_{1\oplus} \Sigma_{n-1} J_{1\oplus}^T + J_{2\oplus} V_n J_{2\oplus}^T \quad (17)$$

The l th of candidate exploration paths is composed of m road-mark points $P_l = \{p_1, p_2, \dots, p_m\}$ and $\delta_d = \|p_m - p_{m-1}\|$; $\delta_\phi = \arctan(\frac{y_{p_m} - y_{p_{m-1}}}{x_{p_m} - x_{p_{m-1}}})$. We use equation (16) and equation (17) to estimate the uncertainty Σ^l as the robot moves along the l th candidate path

$$m^* = \arg \min_m \frac{1}{2} (\Sigma_m^l - \Sigma_{\max})^2 \quad (18)$$

By evaluating the uncertainty of the l th candidate path, a subpath $P'_l = \{p_1, p_2, \dots, p_{m^*}\}$, $1 \leq m^* \leq m$ is generated from the path P_l according to equation (18). The robot explores the scene along the subpath P'_l , the uncertainty estimate Σ_{m^*} of the robot's location does not exceed the threshold Σ_{\max} . That is

$$\det(\Sigma_{m^*}) \leq \det(\Sigma_{\max}), \forall m, 1 \leq m \leq m^* \quad (19)$$

Equation (19) means that if the uncertainty constraint corrects the candidate path, the uncertainty of the robot's location is guaranteed not to exceed Σ_{\max} .

Exploration path evaluation and selection

Information gain (IG) metrics for evaluation

We use information gain as metrics to evaluate each candidate path. The information gain in this article is measured in the number of grids which are covered by a candidate path on the information entropy map I_e . The amount of information gain that can be obtained by exploring along a path is an essential consideration in evaluating candidate paths.

Because we maintain a map of information entropy synchronously with the update of the occupied map, the calculation of the information entropy gain becomes very straightforward. For a certain path $P = \{p_1, p_2, \dots, p_k\}$, the information entropy gain E_g along the path is defined as

$$E_g = \sum_{p_i \in P} \sum_{c \in \text{Region}(p_i)} H'(c) - H(c) \quad (20)$$

where $H(c)$ is the information entropy of a grid cell $c \in \mathbb{R}^2$ which is in the region of radius R_e around the road-mark point p_i . $H'(c)$ indicates the information entropy of this area updated.

$$H'(c) = P(c) \ln P(c) - (1 - P(c)) \ln(1 - P(c)) \quad (21)$$

Reexplore metric for evaluation

When the uncertainty of the robot positioning increases to a critical value, continuing to explore along the path of high entropy may result in the loss of positioning. Exploring alone candidate paths that are in the previously explored area will lead to a meager information entropy gain, but reexploring along these paths can improve the accuracy of the map and reduce the uncertainty of the robot's location

Algorithm 1. Mobile robot autonomous exploration path planning algorithm.

```

1: while (True) do
2:   Update the grid map  $I_g$  and the information entropy
   map  $I_e$ ;
3:   Extract boundaries  $B_e$  and  $B_g$ ;
4:   Construct the graph  $G$  from the grid map;
5:   for each  $x_i \in B_e \cup B_g$  do
6:     for each  $x_j \in Neighbour(x_i)$  do
7:       if  $x_i \in B_e$  then
8:          $W_{x_i, x_j} = k_1 * W_{x_i, x_j}$ ;
9:       else
10:         $W_{x_i, x_j} = k_2 * W_{x_i, x_j}$ ;
11:       end if
12:     end for
13:   end for
14:    $P_{GCP} = GenCandidatePaths(I_g, I_e, B_e, B_g)$ ;
15:    $P = P_{GCP}$ 
16:   for each  $p_k \in P$  do
17:      $p_k = CorrectByUncertaintyEst(p_k)$ ;
18:   end for
19:    $p^* = PathWithMaxInformationGain(P)$ ;
20:   if  $IsEmpty(p^*)$  is True then
21:      $p^* = PathWithMaxReExpCovrage(P_{GCP})$ ;
22:   end if
23:   if  $AvgEntropy(I_g)$  lower than the threshold  $E_{min}$ 
   then
24:     break;
25:   end if
26:    $SendExplorationPath(p^*)$ ;
27: end while

```

$$E_r = \sum_{p_i \in P} \sum_{c \in Region(p_i)} H_0 - H(c) \quad (22)$$

Here we assume that the updated information entropy is the same as the predetermined entropy H_0 to calculate a virtual information entropy gain. $H(c)$ is the information entropy of the grid cell c , and E_r is the obtained virtual information entropy gain.

Comprehensive evaluation and selection algorithm

The exploration path planning method proposed in this article is listed in Algorithm 1. The algorithm uses a grid map I_g and an information entropy map I_e as its input. The grid map used in Algorithm 1 is generated from the GMapping, and the information entropy map is gradually updated as the exploration progresses. The entire exploration process can be roughly divided into three steps.

First of all, the information entropy map is initialized as the same size as the grid map and expand as the grid map. We use the convolution kernel described in the fourth section to convolve the information entropy map I_e and grid map I_g , and then we get the boundary B_e and B_g respectively according to equation (4).

Then we build a graph G from the grid map I_g . The weight of the edges connected to nodes at the boundary will be reduced. The *Dijkstra* algorithm is applied to generate the candidate paths P_{GCP} from the robot position to every grid lying on the boundary. After getting these candidate paths, we first use the uncertainty prediction method in the fourth section to correct the candidate paths. After the above processing, if no candidate path is generated, We will use function *PathWithMaxReExpCovrage()* to implement the method to select a path from the candidate path P_{GCP} .

Finally, we first select the path with the highest information entropy gain from these paths. If the path does not exist, this indicates that we have explored all the areas, or according to the uncertainty constraints. If we continue to explore the unknown area, we will lose the location when a decrease in the average information entropy within the specified E_{min} , the entire exploration process is completed.

Simulations and experiments

We compared the proposed algorithm with two other methods^{11,29} in the simulation. We also verified our algorithm's adaptability and efficiency in three real-world experiments.

Evaluation in synthetic scenes

We use ROS-Stage package⁴⁷ and three public grid maps to set up the simulation environment. The robot used in the simulation is equipped with an odometric sensor with noise covariance V_n and a single-line laser with a maximum detection range of 15 m and a field of view of 190 degrees, and its maximum movement speed is set to 0.25 m/s. The initial uncertainty of the robot pose estimate is set to Σ_0 . All methods are evaluated in the same simulation environment. The first exploration approach is an active SLAM exploration method that trains a GP to predict the maximum information gain under control and use Bayesian optimization to get the best exploration target.²⁹ Another exploration method we compared with is the classic frontier-based approach.¹¹

Figure 6 shows the generated map in the simulation and the recorded exploration trajectory. Figure 7 shows the information entropy changes with exploration. Table 1 represents all statistical results. It gives the total reduction of the *Entropy*, the final exploration *Distance*, and the entropy reduced nats per meter (E_r), the standard deviation of the entropy reducing rate $Std(E_r)$, and the total area covered during the exploration process.

We performed experiments in the same simulation environment. When the robot moves forward 10 cm, we calculate and save the information entropy of the map, the pose of the robot, the covariance matrix E of the location, and the simulation time. We use the information entropy of the map over the traveled distance to evaluate the performance of each algorithm.

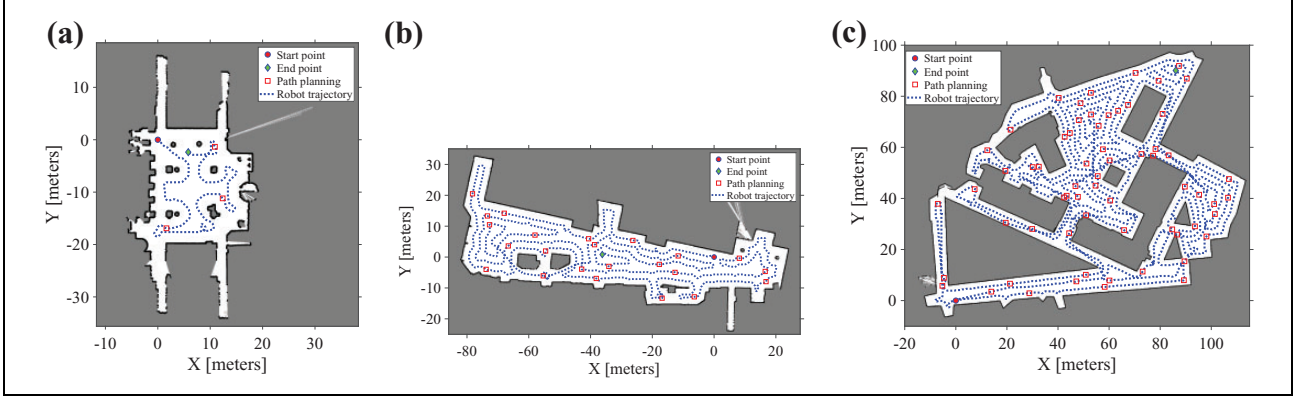


Figure 6. The comparison experiments with the other two methods on three different open-source map. The picture on the left shows the experiment on the *casia* map, the middle represents the experiment on the *fr101* map, and the one on the right describes the experiment on *fr-campus*.⁴⁶

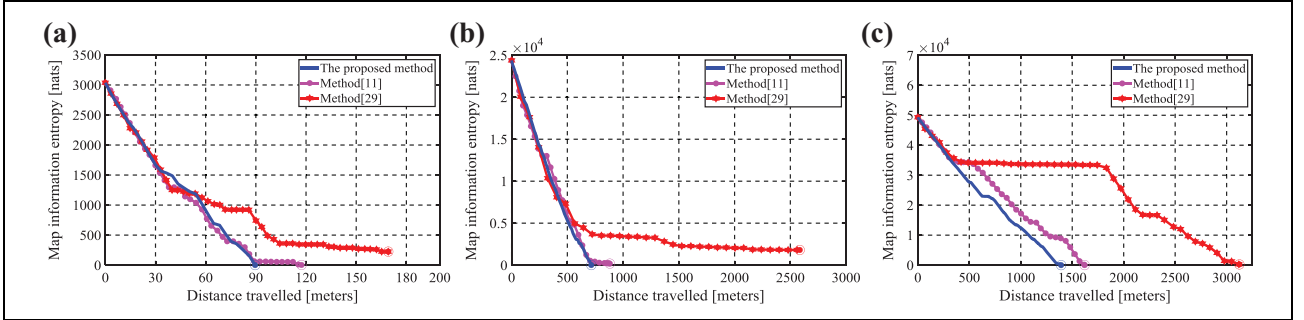


Figure 7. The experimental result corresponding to Figure 6. The curve represents the change of the information entropy of the environment when different exploration methods are adopted. The blue line (the proposed method) shows the shortest exploration path length in three different scenarios, and the information entropy of the map decreases faster throughout the process.

Table 1. Comparative data in the three simulation experiments.

Map	Method	Entropy (nats)	Distance (m)	E_r (nats/m)	Std(E_r)	Coverage area (m ²)
casia (a)	Method ¹¹	3064	117.30	26.12	16.61	191.50
	Method ²⁹	2812	169.16	16.62	17.11	175.75
	Ours	3231	89.57	36.07	9.49	201.94
fr101 (b)	Method ¹¹	24,213	881.06	27.48	14.07	1513.31
	Method ²⁹	22,586	1009.23	22.38	19.28	1411.63
	Ours	25,179	714.85	35.22	7.09	1573.69
fr-campus (c)	Method ¹¹	62,761	1620.33	38.73	13.58	3922.56
	Method ²⁹	49,097	3123.94	15.72	17.34	3068.56
	Ours	64,474	1391.87	46.32	9.20	4029.63

Bold face value indicates that the proposed method is better or has a greater improvement than the other two methods.

In the beginning, the three methods have a similar exploration efficiency. We can see that the decline rate of information entropy is very close to Figure 7. As the exploration progresses, the unexplored area using the comparative methods is broken into pieces, and as a result of the efficiency of the exploration using other methods^{11,29} has dropped a lot. This can be seen from different experiments is shown in Figure 7(a) to (c). As the fragmentation of the unexplored area is aggravated, the maximum information gain under control is becoming

harder to predict by using *Bayesian* optimization.²⁹ This leads to a significant decline in the efficiency of exploration. When the unexplored area is broken into pieces, the robot using the classic frontier-based method wanders in some small but far apart frontier-fragments. This also leads to the same decline in the efficiency of exploration, especially in the final stage. From Figure 7, the exploration efficiency of the methods^{11,29} drops more significantly as the complexity of the environment increases.

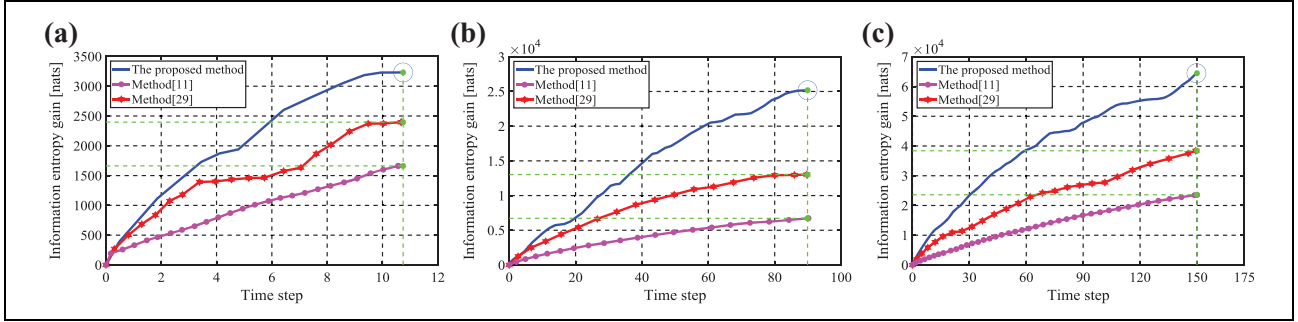


Figure 8. Comparison of the computational speed of the proposed method, the method,¹¹ and the method,²⁹ measured by information entropy gain over time step in the three simulation experiments. Figure (a) shows the experiment on the *casia* map, Figure (b) represents the experiment on the *fr101* map, and Figure (c) describes the experiment on *fr-campus*.⁴⁶

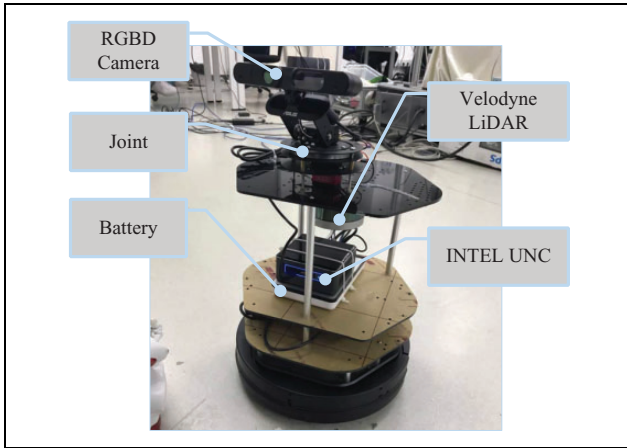


Figure 9. The structure of the real experiment robot. It is built on a Turtlebot robot, with Velodyne VLP-16 Lidar and an Intel UNC mini PC installed on it.

The proposed method is to explore along the boundary instead of going to the border to explore. Because the boundary-based exploration method avoids the fragmentation of the unexplored area, the efficiency of the exploration can be very and efficient constant. Table 1 shows that the average decline rate of E_r in environmental information entropy is higher compared to the others. The standard deviation of the rate $\text{Std}(E_r)$ is smaller than the others. Table 1 also shows that our approach achieves the best coverage (4029.63 m², 201.94 m², 1537.69 m²) of the environment with the shortest exploration path (1391.87 m, 89.57 m, 714.85 m).

Under the same conditions, we conduct comparative experiments of the proposed method with the classical frontier-based approach¹¹ and the Bayesian optimization-based approach.²⁹ Figure 8 provides a comparison plot of information entropy gain over the simulation step in three simulations, respectively. The blue circle indicates the end of the exploration. As the exploration progresses, the frontier-based exploration strategy intensifies the fragmentation of unexplored areas and has a significant impact on the efficiency of exploration. As a result, we can see that the

computational speed of the method¹¹ is the lowest in the three simulations. The method²⁹ is a Bayesian optimization-based method. At each step, it needs to solve a Bayesian optimization problem to find the exploration target. The experimental results show that the proposed method has higher computational efficiency than other methods.

Evaluation in real scene

We also verify our proposed method in real scenes. The robot used in the experiment is a *kobuki* robot, which is equipped with a Velodyne VLP16 multiline lidar, an RDB-D camera, and an odometer. It is shown in Figure 9. The laser sensor scans the surrounding environment at a rate of 10 Hz per second. The odometer module collects information at a rate of 25 Hz per second. The driver and control system software is developed using C++ language on a Linux system with an i7 4-core CPU and 8 GB RAM. The grid map was built using the modified GMapping package under ROS. The algorithm for updating the information entropy map, extracting the boundary, and planning the selection of the exploration path is listed in Algorithm 1.

Figure 10 shows the exploration trajectory, and the generated 3-D point cloud map in the three different real scenes. In the experiment of Figure 10(a) and (b), we set $\alpha = 1.5$, $R_e = 0.75$ m, and the resolution of the grid map is 0.05 m/grid. In the experiment shown in Figure 10(c), we set a different value of $\alpha = 1.0$. Compared to Figure 10(a) and (b), the point cloud generated in Figure 10(c) is denser. Because we chose the smaller parameter $\alpha = 1.0$, which makes more overlap between adjacent observations, and the equivalent S_{dpi} is higher than average, so we can get more scan points.

Since the proposed method is based on the path planning method, it is possible to dynamically adjust the exploration trajectory according to changes in an environment. For a dynamic object, the exploration path does not change as long as it moves outside the safe area of the robot. The exploration algorithm is triggered to replan the exploration path only when a dynamic object enters the safe area of the robot. When the dynamic object is removed, the area where the previous dynamic object located will also be considered

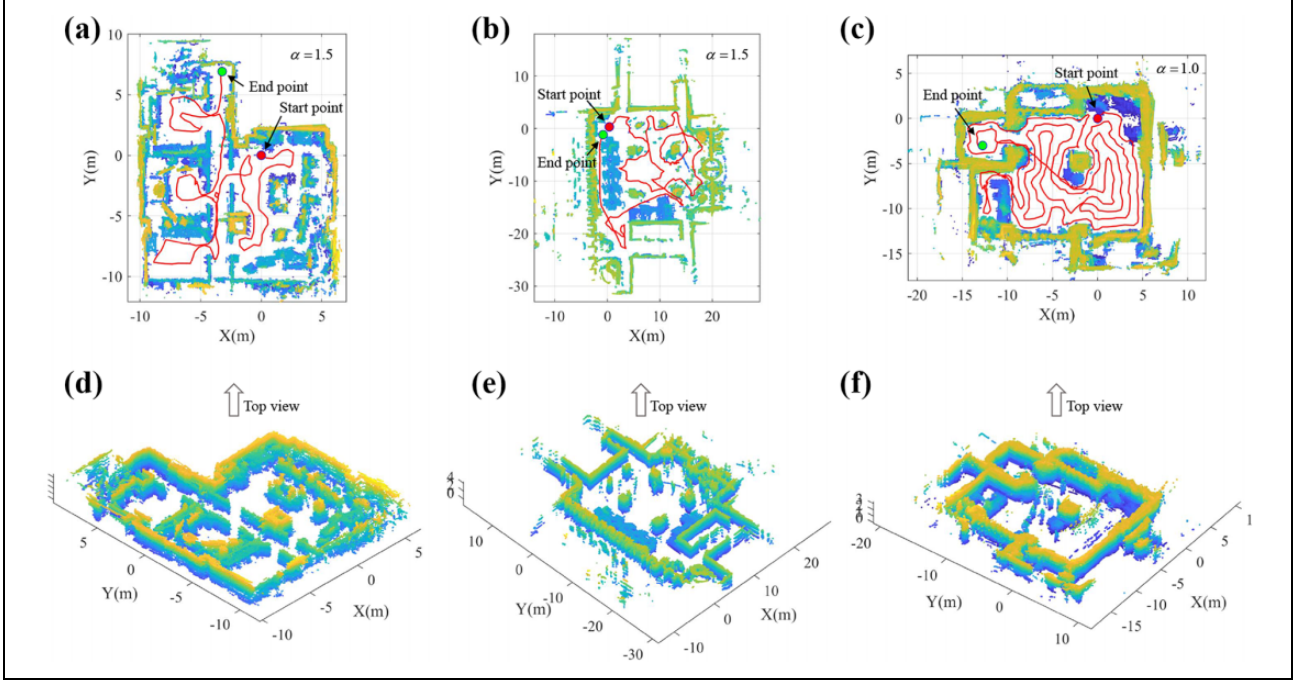


Figure 10. The figure shows the exploration trajectory and the generated 3-D point cloud map after the robot actively explores the three different real scenes. Figure (a), Figure (b), and Figure (c) are top views of Figure (d), Figure (e), and Figure (f), respectively. We choose $\alpha = 1.5$ in Figure (a) and Figure (b), but to get a denser point cloud map, we set α to a smaller value of 1.0 in Figure (c).

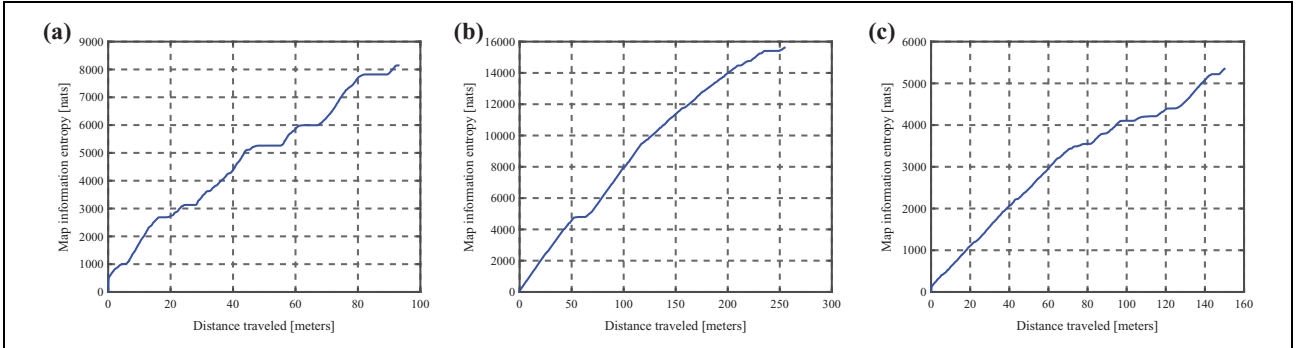


Figure 11. The above figures show the information entropy gain of the map over the traveled distance.

again in the future exploration path. In Figure 10(e) and (f), we can see that there are many pedestrians in the point cloud being built, but from the exploration trajectory in the above picture, we can hardly see the impact of these pedestrians.

Figure 11 shows the information entropy gain of the map over the traveled distance recorded at a specific distance interval during the exploration process. Table 2 lists the experiment results. The proposed method has a relatively stable rate of information entropy reduction in different scenarios.

We also study the effects of uncertainty constraints on the robot's exploration. From Figure 12, we can see that after adding the uncertainty constraint, the uncertainty of the localization of the robot is limited to a specific range, which significantly improves the adaptability of the algorithm to the environment.

Table 2. Comparative data in the three real experiments.

Scenes	Running time (min)	Distance (m)	Coverage	
			area (m^2)	IG (nats)
Real scenes (a)	11.41	93.32	102.21	7795
Real scenes (b)	14.41	150.36	358.81	4242
Real scenes (c)	21.50	255.45	230.33	17024

IG: Information gain.

Conclusion

In this article, we address the exploration path planning for a robot to achieve SLAM in an unknown environment. We proposed an exploration path planning method, which is based on boundary features and uncertainty estimations.

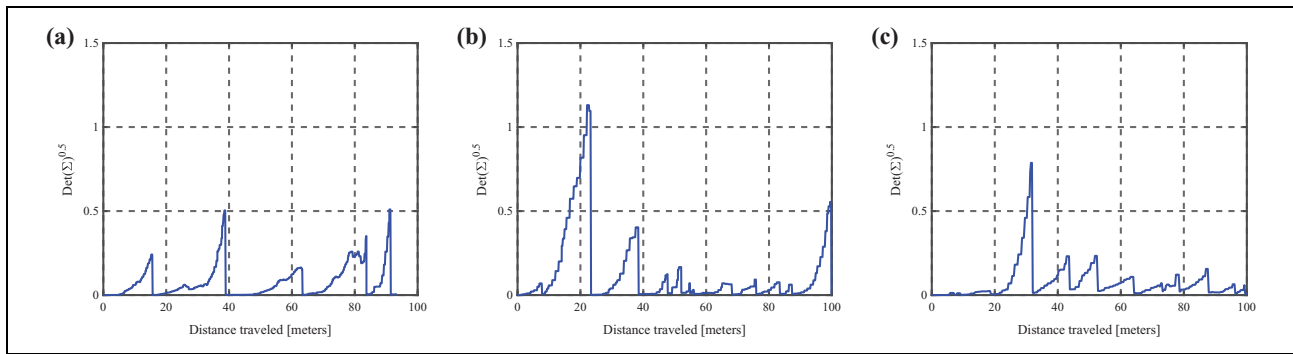


Figure 12. It shows the changes in the uncertainty of the robot's pose in the three real environments using the proposed method.

The proposed path planning method tries to generate candidate paths along the boundaries. It avoids breaking the unexplored region into pieces and can be adjusted by α to achieve a higher exploration rate or denser mapping. We use dead reckoning to estimate the uncertainty of the robot's localization along each candidate path and limit the uncertainty within a given threshold. The information entropy gain and the uncertainty estimation are simultaneously considered to trading off exploration against exploitation. Then, a path selection strategy is used to generate the optimal exploration path from these candidate paths. Three synthetic scenes and three real scenes with active SLAM tasks are considered in the simulations and experiments. The results show that the proposed method has a better performance compared to two classic methods.

In our further work, we will perform more real-world experiments, and verify the ability of the proposed approach in an outdoor environment. We will study the ways for multiple robots to share perceptual information and study the mechanism of multi-robot cooperative exploration and then extend the proposed method to multiple robots to improve the overall exploration efficiency.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported in part by the National Natural Science Foundation of China under Grant U1713222, 61773378, and 61703401; and in part by the Foundation for Innovative Research Groups of the National Natural Science Foundation of China under Grant 61421004, and in part by Beijing Science and Technology Project under Grant Z181100003118006.

ORCID iD

Peng Li  <https://orcid.org/0000-0002-4992-1295>

Supplemental material

Supplemental material for this article is available online.

References

1. Davison AJ, Reid ID, Molton N, et al. MonoSLAM: real-time single camera SLAM. *IEEE Trans Pattern Anal Mach Intell* 2007; 29(6): 1052–1067.
2. Engel J, Schops T, and Cremers D. LSD-SLAM: large-scale direct monocular SLAM. In: *Computer vision—ECCV 2014*, Zurich, Switzerland, 6–12 September 2014, pp. 834–849.
3. Hess W, Kohler D, Rapp HH, et al. Real-time loop closure in 2D LIDAR SLAM. In: *2016 IEEE international conference on robotics and automation (ICRA)*, Stockholm, Sweden, 16–21 May 2016, pp. 2494–2499. IEEE.
4. Klein G and Murray DW. Parallel tracking and mapping for small AR workspaces. In: *Proceedings of the 2007 6th IEEE and ACM international symposium on mixed and augmented reality*, Nara, Japan, 13–16 November 2007.
5. Murartal R, Montiel JMM, and Tardos JD. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Trans Robot* 2015; 31(5): 1147–1163.
6. Whelan T, Leutenegger S, Salasmoreno RF, et al. Elastic fusion: real-time dense SLAM and light source estimation. *Int J Robot Res* 2015; 35(14): 1697–1716.
7. Li D and Gao H. A hardware platform framework for an intelligent vehicle based on a driving brain. *Engineering* 2018; 4(4): 464–470.
8. Gao H, Yu H, Xie G, et al. Hardware and software architecture of intelligent vehicles and road verification in typical traffic scenarios. *IET Intell Trans Syst* 2019; 13(6): 960–966.
9. Gao H, Cheng B, Wang J, et al. Object classification using CNN-based fusion of vision and LIDAR in autonomous vehicle environment. *IEEE Trans Ind Informat* 2018; 14(9): 4224–4231.
10. Wang Y, Wang R, Wang S, et al. Underwater bio-inspired propulsion: from inspection to manipulation. *IEEE Trans Ind Electron*. DOI: 10.1109/TIE.2019.2944082, 2019.
11. Yamauchi B. A frontier-based approach for autonomous exploration. In: *1997 IEEE international symposium on computational intelligence in robotics and automation CIRA'97*, Monterey, CA, USA, 10–11 July 1997, pp. 146–151. IEEE.
12. Feder HJS, Leonard JJ, and Smith CM. Adaptive mobile robot navigation and mapping. *Int J Robot Res* 1999; 18(7): 650–668.

13. Stachniss C and Burgard W. Exploring unknown environments with mobile robots using coverage maps. In: *International joint conference on artificial intelligence (IJCAI)*, Acapulco, Mexico, 9–15 August, 2003, pp. 1127–1132.
14. Freda L and Oriolo G. Frontier-based probabilistic strategies for sensor-based exploration. In: *2005 IEEE international conference on robotics and automation (ICRA)*, Barcelona, Spain, 18–22 April 2005, pp. 3881–3887. IEEE.
15. Bourgault F, Makarenko A, Williams SB, et al. Information based adaptive robotic exploration. In: *2002 IEEE/RSJ international conference on intelligent robots and systems (IROS 2002)*, Lausanne, Switzerland, 30 September–4 October 2002, pp. 540–545. IEEE.
16. Carlone L, Du J, Ng MEK, et al. Active SLAM and exploration with particle filters using Kullback-Leibler divergence. *J Intell Robot Syst* 2014; 75(2): 291–311.
17. Carrillo H, Latif Y, Rodriguez-Arevalo ML, et al. On the monotonicity of optimality criteria during exploration in active SLAM. In: *2015 IEEE international conference on robotics and automation (ICRA)*, Seattle, WA, USA, 26–30 May 2015, pp. 1476–1483. IEEE.
18. Chaves SM and Eustice RM. Efficient planning with the Bayes tree for active SLAM. In: *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS 2016)*, Daejeon, South Korea, 9–14 October 2016, pp. 4664–4671. IEEE.
19. Deng X, Zhang Z, Sintov A, et al. Feature-constrained active Visual SLAM for mobile robot navigation. In: *2018 IEEE international conference on robotics and automation (ICRA)*, Brisbane, QLD, Australia, 21–25 May 2018, pp. 7233–7238. IEEE.
20. Khaksar W, Hong TS, Khaksar M, et al. Sampling-based tabu search approach for online path planning. *Adv Robot* 2012; 26(8–9): 1013–1034.
21. Leung C, Huang S, and Dissanayake G. Active SLAM in structured environments. In: *2008 IEEE international conference on robotics and automation*, Pasadena, CA, USA, 19–23 May 2008, pp. 1898–1903. IEEE.
22. Potthast C and Sukhatme GS. A probabilistic framework for next best view estimation in a cluttered environment. *J Vis Commun Image R* 2014; 25(1): 148–164.
23. Xu K, Zheng L, Yan Z, et al. Autonomous reconstruction of unknown indoor scenes guided by time-varying tensor fields. *ACM Trans Graph (TOG)* 2017; 36(6): 202.
24. Liu L, Xia X, Sun H, et al. Object-aware guidance for autonomous scene reconstruction. *ACM Trans Graph (TOG)* 2018; 37(4): 104.
25. Mobarhani A, Nazari S, Tamjidi A, et al. Histogram based frontier exploration. In: *2011 IEEE/RSJ international conference on intelligent robots and systems*, San Francisco, CA, USA, 25–30 September 2011, pp. 1128–1133. IEEE.
26. Leung C, Huang S, and Dissanayake G. Active SLAM using model predictive control and attractor based exploration. In: *2006 IEEE/RSJ international conference on intelligent robots and systems*, Beijing, China, 9–15 October 2006, pp. 5026–5031. IEEE.
27. Liu Y, Sun F, Tao T, et al. A solution to active simultaneous localization and mapping problem based on optimal control. In: *2007 international conference on mechatronics and automation*, Harbin, China, 5–8 August 2007, pp. 314–319. IEEE.
28. Vallvé J and Andradecetto J. Active pose SLAM with RRT. In: *2015 IEEE international conference on robotics and automation (ICRA)*, Seattle, WA, USA, 26–30 May 2015, pp. 1382–1388. IEEE.
29. Bai S, Wang J, Chen F, et al. Information-theoretic exploration with Bayesian optimization. In: *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS 2016)*, Daejeon, South Korea, 9–14 October 2016, pp. 4664–4671. IEEE.
30. Souza JR, Marchant R, Ott L, et al. Bayesian optimisation for active perception and smooth navigation. In: *2014 IEEE international conference on robotics and automation (ICRA)*, Hong Kong, China, 31 May–7 June 2014, pp. 5558–5564. IEEE.
31. Maurovic I, Seder M, Lenac K, et al. Path planning for active SLAM based on the D* algorithm with negative edge weights. *IEEE Trans Syst Man Cyber Syst* 2018; 48(8): 1321–1331.
32. Valencia R, Miro JV, Dissanayake G, et al. Active pose SLAM. In: *2012 IEEE/RSJ international conference on intelligent robots and systems*, Vilamoura, Portugal, 7–12 October 2012, pp. 1885–1891. IEEE.
33. Kim A and Eustice RM. Perception-driven navigation: active visual SLAM for robotic area coverage. In: *2013 IEEE international conference on robotics and automation (ICRA)*, Karlsruhe, Germany, 6–10 May 2013, pp. 4271–4278. IEEE.
34. Meng Z, Qin H, Chen Z, et al. A two-stage optimized next-view planning framework for 3-D unknown environment exploration, and structural reconstruction. *IEEE Robot Automat Lett* 2017; 2(3): 1680–1687.
35. Meng Z, Sun H, Qin H, et al. Intelligent robotic system for autonomous exploration and active SLAM in unknown environments. In: *2017 IEEE/SICE international symposium on system integration (SII)*, Taipei, Taiwan, 11–14 December 2017, pp. 651–656.
36. Sim R and Roy N. Global A-optimal robot exploration in SLAM. In: *2005 IEEE international conference on robotics and automation (ICRA)*, Barcelona, Spain, 18–22 April 2005, pp. 661–666. IEEE.
37. Stachniss C, Grisetti G, and Burgard W. Information gain-based exploration using rao-blackwellized particle filters. *Proceed Robot Sci Syst* 2005; 01: 65–72.
38. Tao T, Huang Y, Sun F, et al. Motion planning for SLAM based on frontier exploration. In: *2007 international conference on mechatronics and automation*, Harbin, China, 5–8 August 2007, pp. 2120–2125. IEEE.
39. Trivun D, Salaka E, Osmankovic D, et al. Active SLAM-based algorithm for autonomous exploration with mobile robot. In: *2015 IEEE International Conference on Industrial Technology (ICIT)*, Seville, Spain, 17–19 March 2015, pp. 74–79. IEEE.

40. Vallvé J and Andradecetto J. Potential information fields for mobile robot exploration. *Robot Autonom Syst* 2015; 69: 68–79.
41. Whaite P and Ferrie FP. Autonomous exploration: driven by uncertainty. *IEEE Trans Patt Anal Mach Intell* 1997; 19(3): 193–205.
42. Yamauchi B. Frontier-based exploration using multiple robots. In: *AGENTS '98 Proceedings of the second international conference on Autonomous agents*, Minneapolis, MN, USA, 10–13 May 1998, pp. 47–53.
43. Mu B, Giamou M, Paull L, et al. Information-based active SLAM via topological feature graphs. In: *2016 IEEE 55th conference on decision and control (CDC)*, Las Vegas, NV, USA, 12–14 December 2016, pp. 5583–5590. IEEE.
44. Grisetti G, Stachniss C, and Burgard W. Improved techniques for grid mapping with Rao-Blackwellized particle filters. *IEEE Trans Robot* 2007; 23(1): 34–46.
45. Rodriguez-Arevalo ML, Neira JL, and Castellanos JA. On the importance of uncertainty representation in active SLAM. *IEEE Trans Robot* 2018; 34(3): 829–834.
46. Hornung A. Octomap 3D scan dataset, <http://ais.informatik.uni-freiburg.de/projects/datasets/octomap/> (accessed 18 February 2019).
47. Gerkey BP, Vaughan RT, and Howard A. Reusable robot software and the player/stage project. *Software Eng Exper Robot* 2007; 30: 317–323.