A Two-Stage Optimized Next-View Planning Framework for 3-D Unknown Environment Exploration, and Structural Reconstruction

Zehui Meng, Hailong Qin, Ziyue Chen, Xudong Chen, Hao Sun, Feng Lin, and Marcelo H. Ang Jr

***Abstract*—In this paper, we present a solution for autonomous exploration and reconstruction in 3-D unknown environments without *a priori* knowledge of the environments. In our framework, a two-stage heuristic information gain-based next-view planning algorithm is performed to dynamically select and update candidate viewpoints, followed by immediate next-best-view extraction and corresponding motion planning. The two-stage planner consists of a frontier-based boundary coverage planner and a fixed start open travelling salesman problem solver, such that the planner returns an exploration path with the consideration of global optimality in the context of accumulated space information. The effectiveness of our work is evaluated with simulation and experimental tests. The results and comparisons prove our system is able to autonomously explore the 3-D unknown environments and reconstruct the structural model with improved exploration efficiency in terms of path quality and total exploration time.**

***Index Terms*—Autonomous agents, reactive and sensor-based planning, SLAM.**

I. Introduction

BEING able to search environments and acquire 3-D data allows many forms of online or offline ambience reconstruction using SLAM [8], [15], [21] to provide necessary information for robot navigation tasks. Conventionally, environment reconstruction works relied on human guidance [1]. Such approaches are effective in terms of system simplicity and ease of implementation, but require plenty of coordination effort and prone to be affected by the human subjectivities and skills. As the applications of self-navigating robots flourish, the functionality of autonomous environment exploration and reconstruction is becoming a trend.

Fig. 1. UAV platforms with rotating laser module employed in this framework. (a) Horizontal Configuration, (b) Vertical Configuration

Researchers have put their effort in solving view planning issues with the aid of a prescribed 2-D map and developed view planning algorithms for 3-D mapping [4], [11], [16], [20], [14]. However, when no a-priori information is available, such algorithms will have difficulty exploring the unknown environments. In this case, strategies like active SLAM make more sense, which serves to build a detailed map for unknown environments without prior knowledge through incremental online calculation of next view position [6]. Up to date, many active SLAM algorithms are attempted for 2-D and 2.5-D mapping only and usually yield locally optimal paths due to their greedy-exploring nature, leaving the 3-D implementation and the problem of global optimality yet to be studied. Therefore, this work aims at combining the advantages of existing next-view planning and active SLAM strategies.

In this paper, we present a novel framework consisting of a platform-independent 2-stage optimized view planner which combines a viewpoints coverage planning with a Fixed Start Open Traveling Sales Man Problem solving, a volumetric map-based navigation interface, and a rotating laser SLAM module (as shown in Fig. 1) for complete exploration and reconstruction of 3-D unknown environments. Simulation and experimental evaluations on both vision-based and laser-based platforms are conducted in comparison with the state-of-the-art work. Our framework is shown to possess the feature of autonomous exploration in 3-D unknown environment without a-priori knowledge while alleviating the drawback of greedy exploration through planning in the global context, providing improved overall exploration efficiency in terms of path quality and total exploration time.

The remainder of the paper is arranged as follows. Section II gives a short review of related literatures followed by the problem description regarding this work in Section III. Section IV presents the essentials of the proposed framework. Section V

Manuscript received September 10, 2016; accepted January 4, 2017. Date of publication January 18, 2017; date of current version May 16, 2017. This paper was recommended for publication by Associate Editor S. Behnke and Editor T. Asfour upon evaluation of the reviewers comments. This work was supported by the collaborative project under Temasek Laboratory and NUS Graduate School of Integrative Science and Engineering, Singapore.

Z. Meng is with the NUS Graduate School of Integrated Science and Engineering, National University of Singapore, Singapore 118998, and also with the Department of Mechanical Engineering, National University of Singapore, Singapore 117411 (e-mail: A0077878@u.nus.edu).

H. Qin, X. Chen, and F. Lin are with the Temasek Laboratory and the Department of Electrical and Computer Engineering, Singapore 117604 (e-mail: mpeqh@nus.edu.sg; E0114868@u.nus.edu; tsllinf@nus.edu.sg).

Z. Chen, H. Sun, and M. H. Ang Jr are with the Department of Mechanical Engineering, National University of Singapore, Singapore 117411 (e-mail: A0091756@u.nus.edu; A0132560@U.NUS.EDU; mpeangh@gmail.com).

This letter has supplementary downloadable material available at http:// ieeexplore.ieee.org, provided by the authors.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LRA.2017.2655144

2377-3766 © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Authorized licensed use limited to: University of Tokyo. Downloaded on June 21,2025 at 07:59:21 UTC from IEEE Xplore. Restrictions apply.

shows the results and discussions of simulations and experiments. The conclusion is given in Section VI.

II. Related Works

As solutions to non-model based dynamic view planning subject to partially known or changing environments, many works have tried to extend the concept of active SLAM and "next-best-views". Kruno [13] presents an active SLAM solution with active loop closing for 2D unknown environments mapping by employing an exploration strategy extended from Ekman's exploration algorithm [9] and "jump edges" resembling the frontier-based sampling technique [22], which is in a similar manner as [7] and [5]. Unfortunately, the authors have not shown any extension or implementation of their work on 3-D environments. Potthast and Sukhatme [17], and Isler *et al.* [12] introduce probabilistic information gain-based variants of the next-best-view problem in which the algorithms focus on data acquisition through iterative estimation of viewpoints and viewing angles that maximize a reward function with an occluded setting, helping extend the active autonomous exploration to 3-D cases with the concept of information gain. More recently, Bircher *et al.* [3] present a path planning algorithm with a iterative receding horizon "next-best-view" scheme for autonomous exploration of unknown space. Their algorithm evaluates an online computed random tree with the potential unmapped space and executes the first edge of the best branch in each iteration. In a similar sense, Yoder and Scherer [23] propose an incremental path planning algorithm using their surface frontier evaluation. Instead of constructing an explicit information gain model, they utilize a simple but effective utility function that maximizes the efficiency of exploring the surface frontier to guide observation of unknown structural surfaces. The above active exploration strategies provide solutions to complete exploration of unknown environment without a-priori information. But their greedy exploration features do not guarantee a globally optimal exploration path, leading to unwanted reciprocating motions and prone to local optima traps (thus unnecessarily longer exploration paths). This can be energy-inefficient in terms of power consumption, especially in large-scale environment exploration scenarios. To account for the global optimality of the exploration path, authors in [2] introduce an alternating 2-step optimization paradigm for structural inspection planning which takes care of both the full coverage problem and the optimal inspection path. The algorithm employs an online iterative viewpoints resampling followed by optimal path updating using a heuristic TSP solver to reduce back-and-forth traveling caused by greedy exploration. The algorithm is evaluated to have high performance for complex structures while respecting vehicle and sensor constraints. However, the algorithm assumes a triangular mesh representation of the structure to be available, which is tough for visibility checking, therefore limiting its generality of application and adaptability towards unknown environments.

In our work, the proposed framework combines the advantages of the above literature and formulates the utilities in a novel manner, delivering an optimized view planning strategy for volumetric environment model construction without a-prior information, while concerning both the problem of full coverage and the global optimality of exploration path.

III. Problem Definition

The problem of optimized view planning, as is considered in this work, comprises of a 3-D unstructured environment to be explored and a mobile robot equipped with integrated sensors. The environment can be continuous or represented by discrete probabilistic grids whose probabilities indicate the possible obstacle regions, free space and residual unknown regions. Let $X \subseteq R^3$ be the state space describing the environment and $X_{\text{free}} \subset X$ be the set of states in free space. The problem looks for a set of admissible viewpoint configurations X_{view} by randomly sampling states from the free space with the uniform random function $\{x \sim \mathcal{U}(X_{\text{free}})\}$ such that the sensors can observe the whole environment. We define $\sigma : [0, 1] \mapsto X_{\text{free}}$ as a path solution consisting of a sequence of states connecting all $x \in X_{\text{view}}$, i.e. to visit all viewpoints at least once. Let S denote the set of all nontrivial solutions. The optimal solution σ^* optimizing the cost $c : S \mapsto R_{\geq 0}$ is pursued so as to lead an optimal path in the free space for complete exploration:

$$\sigma^* = \underset{\sigma \in S}{\operatorname{argmin}} \{c(\sigma) | \forall x \in X_{\text{view}}, \exists \sigma(t) = x, \forall t \in [0, 1], \sigma(t) \in X_{\text{free}}\} \quad (1)$$

where $R_{\geq 0}$ denotes real numbers that are non-negative.

Note that quality measures for the planned paths should be situation specific, depending on the optimization objectives.

Algorithm 1: `OptimizedNextViewPlanner`.

1: *Initial scan and map update (**Free**,* **Occupied**, **Unknown***)*
2: *Coverage sampling using frontier sampler (Algorithm 2-3)*
3: *Prune the viewpoints if map updated (Algorithm 4)*
4: *Calculate the cost matrix for the* **FSOTSP** *solver (Algorithm 5)*
5: *Solve the* **FSOTSP** *for exploration sequence (Algorithm 5)*
6: *Engage the volumetric navigation interface*
7: *Compute valid motion plan for the first edge extracted from the exploration sequence*
8: **while** $termination\ condition\ not\ triggered$ **do**
9: *Update the environment model*
10: *Repeat step 2–7*
11: **end while**

IV. Proposed Framework

The proposed framework consists of a platform-independent optimized view planning algorithm, a rotating laser-based 3-D SLAM module, and a volumetric map-based navigation interface. The system integration is built in ROS. Dynamic online view planning and the associated motion planning are performed to explore the unknown environment and update the 3-D OctoMap obtained using our SLAM module, while respecting the kinematic and sensor constraints (e.g., field of view (FoV), sensing range, etc.) of the robotic platform.

An overview of our framework is shown in Fig. 2 and the algorithmic implementation is summarized in Algorithm 1. When entering an unknown environment, the system performs an initial scan and mapping. Based on the initial map, the planner generates a minimal set of candidate viewpoints using our novel frontier-based coverage sampler to cover the boundary of the explored free space and the unknown space. During sampling, the volumetric map consecutively marks the visible neighbourhood of the samples such that the predicted observations of

Authorized licensed use limited to: University of Tokyo. Downloaded on June 21,2025 at 07:59:21 UTC from IEEE Xplore. Restrictions apply.

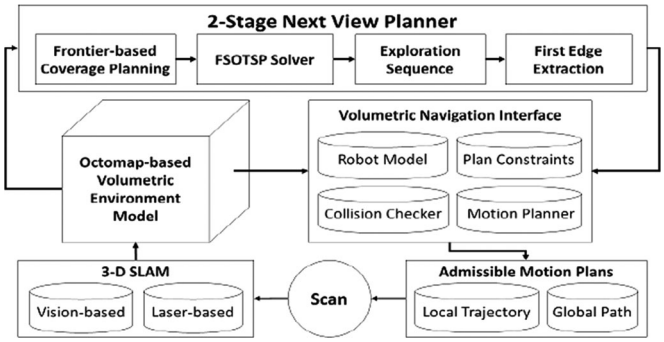

Fig. 2. The overall system architecture of the proposed framework.

subsequent samples do not extensively overlap with existing viewpoints. The candidates are pruned every iteration with the dynamically updated environment model. We would like the agent to traverse the viewpoints with a globally optimal path so that the environment is explored with the shortest travel and least back-and-forth motion. This is achieved by employing a fixed-start open travelling salesman problem (FSOTSP) solver that heuristically computes an optimal open exploration sequence. The first edge of the resulting exploration sequence is extracted and the corresponding path is validated with collision checking and local motion generation (using CHOMP [18]) through our volumetric navigation interface regarding the map constructed via online SLAM. Such steps are iteratively executed to acquire new information and update the environment model. Details of the framework are elaborated as below.

A. 2-Stage Optimized Next View Planning

The 2-stage optimized view planner contains two major components, a frontier-based coverage planning process and FSOTSP solver, iteratively solving for optimal exploration viewpoints sequences regarding the instant environment model (in our case a volumetric representation built with OctoMap).

1) Volumetric Information Gain Model: This next view planning algorithm is based on a volumetric information gain model. The information gain is defined close to the entropy described in [17], but the implementation in this model is approximated with a simplified heuristic measure that reflects the potential increments of knowledge. The indicator measure is formulated such that the hidden information in unobserved cells is appreciated rather than observed cells. Instead of employing a probabilistic distribution with Hidden Markov Model, the information gain is estimated as potential knowledge increment from a visibility-based propagation with ray-casting, regarding the sensor model.

Let $o_i \in [0,1]$ stand for the occupancy state of the i^{th} volumetric cell on the ray. The information gain at o_i observed from a viewpoint $x_{\text{view}} \in X_{\text{view}}$ is defined as:

$$IG(p(o_i|x_{\text{view}},z)) = H(p(o_i)) - H(p(o_i|x_{\text{view}},z)) \quad (2)$$

where $H(p(o_i))$ and $H(p(o_i|x_{\text{view}},z)$ stand for the entropy of a cell occupancy probability before and after integration of measurement z at x_{view}, respectively.

The expected total information gain of x_{view} is then calculated by integrating all possible measurements:

$$E[IG(x_{xiew})] = \int_z p(z|x_{\text{view}}) \sum_{o_i \in C(z)} IG(p(o_i|x_{\text{view}},z))dz \quad (3)$$

where z is the measurement performed over the map and $C(z)$ is the set of covered cells by the measurement.

For implementation considerations, we propose a simplified heuristic indirect indicator of the information gain instead of calculating a closed-form solution which is practically hard. The heuristic gain (denoted by $\widehat{g}$) builds on the occupancy probability and a visibility-based propagation model to reflect change of uncertainty of the occupancy probability through exploring uncertain regions:

$$\widehat{g}(p(o_i|x_{\text{xiew}},z)) = p(o_i|x_{\text{view}},z)p(v_i|o_{0:i-1},x_{\text{view}},z) \quad (4)$$

where $p(o_i|x_{\text{view}},z)$ and $p(v_i|o_{0:i-1},x_{\text{view}},z)$ are the current occupancy probability as registered in the instantaneous map and the visibility probability of o_i measured from the viewpoint x_{view}, respectively. Condition x_{view}, z indicates observation z to be made at x_{view}. The visibility probability is related to previous occupancy probabilities traversed by the ray to reach the current cell. The visibility transition can be naturally propagated as a function of the unoccupancy probability (complement of occupancy probability) distribution. A simple function can be the product of the unoccupancies:

$$p(v_i|o_{0:i-1},x_{\text{view}}) = \prod_{j=0}^{i-1}(1-p(o_j|x_{\text{view}})) \quad (5)$$

which is compliant with the fact that higher occupancy indicates larger likelihood of occlusion, thus smaller visibility. Obviously, free cells and occupied cells end up in small values of the heuristic product while unknown cells in greater values, as the parabolic shape suggests on interval [0,1].

The total heuristic gain at x_{view} is then calculated as:

$$E[\widehat{G}_{x_{\text{view}}}] = \int_z p(z|x_{\text{view}}) \sum_{o_i \in C(z)} \widehat{g}(p(o_i|x_{\text{view}}))dz \quad (6)$$

with z being the measurement in the volumetric map and $C(z)$ the set of cells covered by the sensor. With such a volumetric information gain model, the information gain is appreciated as major contribution from unknown spaces.

2) Frontier-Based Coverage Planning (Algorithms 2-3): When the planner is called, a minimum set of viewpoints is sampled from the instant environment model to cover the frontier boundary regions between the explored free space and the unknown space. This is realized by uniformly generating a pair of neighbouring samples and checking for their occupancy as described in Algorithms 2-3. The valid sample of this pair is selected as a candidate viewpoint if and only if: 1. they are visible to each other; 2. one of them is $Free$ while its couple is $Unknown$; 3. the $Free$ sample does not fall in the predicted observation of previous viewpoints. In other words, the viewpoint has to be a free sample near the frontier boundary regions yet to be explored and able to seen an unknown sample.

We create an attribute in the Octree structure employed in the Octomap to address the predicted observation coverage of the viewpoints. Every time a candidate viewpoint is selected,

Authorized licensed use limited to: University of Tokyo. Downloaded on June 21,2025 at 07:59:21 UTC from IEEE Xplore. Restrictions apply.

Algorithm 2: `FrontierSampler` $(X_{samples})$.

while $No.OfSamples < Max.No.OfSamples$ && $!FullCoverage$ **do**
  $x_1 \leftarrow$ `RandomSample`$(\mathcal{U}(X))$
  $x_2 \leftarrow$ `RandomNeighbourhoodSample`(x_1, r)
  if `FrontierChecker`(x_1, x_2) **then**
    $x_{found} \leftarrow$ `TheValidOf`(x_1, x_2)
    if !`CheckMarked`(x_{found}) **then**
      $X_{samples} \xleftarrow{+} \{x_{found}\}$
      `CalculateGain`(x_{found})
      `MarkCovered`(x_{found})
    end if
  end if
end while

Algorithm 3: `FrontierChecker` (x_1, x_2).

if `GetOccupancy`(x_1) and `GetOccupancy`(x_2) have one being $Free$ and the other being $Unknown$ **then**
  if `CheckVisibile`(x_1, x_2) **then**
    return $True$
  end if
end if
return $False$

its surrounding space is engaged for information gain calculation in accordance with the sensor model (regarding the robot configuration, FoV and range of the sensor). Ray-casting is performed on the volumetric map to roll over the cells observable by the sensor and aggregate the information gain obtained from the volumetric model. During ray-casting, the attribute of all the cells visible from the viewpoint are marked as true by the predicted observation (`MarkCovered` function in Algorithm 2) so that they are excluded from subsequent sampling in order to avoid unnecessary sensor coverage overlapping (`CheckMarked` function in Algorithm 2). The sampling is repeatedly performed until the frontier boundary regions of the instantaneous map are well covered. This is indicated by the sampler not returning any samples over a specified time period which is scaling proportionally to the unmapped volume within the specified bounding exploration volume.

3) Viewpoints Pruning (Algorithm 4): As the map updates dynamically, the information gains of the remaining viewpoints (edges of which are yet to be executed) need to be re-evaluated because their information gain may have been affected by the most recent scan. These remaining viewpoints together with the newly inserted samples form a new set of candidate viewpoints. To maintain a reasonable (necessary but not excessive) exploration coverage, viewpoints whose information gain drops below a specified threshold will be pruned as seen in Algorithm 4, so that the size of the viewpoints set is kept computationally efficient for the FSOTSP solver.

4) FSOTSP Solver (Algorithm 5): Differing from the conventional traveling salesman problem (TSP) which requires a closed-loop tour, we consider an optimization regarding its variant - a Fixed Start Open TSP (FSOTSP) that requires only an open-loop path starting from a fixed viewpoint and covering all the rest viewpoints. Furthermore, we employ a heuristic for the cost function engaged in the FSOTSP solver to accelerate

Algorithm 4: `Prune` $(X_{samples})$.

for $i = 0$ to $i <$ `SizeOf`$(X_{samples})$ **do**
  `UpdateGain`(x_i)
end for
$X_{samples} \xleftarrow{-} \{x \in X_{samples} | x.gain < CutoffThreshold\}$

Algorithm 5: `FSOTSP Solver` $(X_{samples})$.

for $i = 0$ to $i <$ `SizeOf`$(X_{samples})$ **do**
  for $j = 0$ to $j <$ `SizeOf`$(X_{samples})$ **do**
    $CostMatrix(i, j) =$ (`Norm`(x_i, x_j)) / $(\widehat{G}_{x_i} + \widehat{G}_{x_j} * \exp^{-\eta \texttt{Norm}(x_i, x_j)})$
  end for
end for
$ExplorationSequence =$ `GA`$(X_{samples}, CostMatrix)$

the optimization process since TSP is computational complex. Instead of employing the actual path length given by the motion plan, we take a Euclidean distance and information gain based heuristic cost which reflects the 'utility' (effort paid for gaining information along a certain edge) of the edges between viewpoints. The cost function is formulated such that the major penalty is applied on traveling distance and is reversely proportional to information gain:

$$C \propto \frac{1}{U} = dist / \left(\widehat{G}_{start} + \widehat{G}_{goal} * \exp^{-\eta \cdot dist} \right) \tag{7}$$

where C, U, $dist$, $\widehat{G}_{start}$ and $\widehat{G}_{goal}$ stand for the cost, the 'utility', the Euclidean distance, and the information gains of the start and goal viewpoints, respectively. Such evaluation emphasizes the attenuation of the 'utility' in the direction of travel by multiplying an exponential depreciation term to the information gain $\widehat{G}_{goal}$ at the goal of travel. The variable $\eta \geq 0$ is a tuning parameter for addressing the influence factor of such depreciation with regard to the distance of travel, therefore used to adjust the trade-off between the information gain and the distance of travel for different optimization objectives. Larger η penalizes more on the distance and forces the 'edges' to be directed in the way that the end with larger information gain is connected with priority (i.e., an edge $\overline{AB}$ is preferred to be traversed from A to B than from B to A if the information gain at A is more promising), while smaller η allows more tolerances to the distance and directions of travel. As the problem size increases, the above heuristic saves reasonable computation effort for calculating valid motion plans.

After the pruning step, the set of candidate viewpoints is input to the solver with the associated cost matrix. Leveraging the heuristic, we implement a Generic Algorithm (GA) approach inspired by the work in [19] to iteratively solve for the (near) optimal sequence of the candidate viewpoints (Algorithm 5). Such approach is preferred because it can provide a competitive speed and high-quality optimization solution. In our implementation, the process of breeding offspring consists of a simplified selection and mutation mechanism and omits the crossover operation. We blender up the viewpoints and input a group of random sequences (routes) as the population. The total cost of

Authorized licensed use limited to: University of Tokyo. Downloaded on June 21,2025 at 07:59:21 UTC from IEEE Xplore. Restrictions apply.

a route serves as the objective function to be optimized. For each population, the best route is sought, whose cost is used to update the best-cost-so-far. The population is then partitioned to form random sub-groups that divide up the population size. The best route in each sub-group is then selected for producing the offspring within its group. The breeding process mutates portions of the selected best route by swapping and sliding positions of elements within the same piece to produce new routes. Repeating such selection and mutation processes until the new routes fills up each sub-group gives a new generation. Finally we compare the best route in the new population with the old population to make sure the population is updated with reduced best-cost-so-far. The whole process is repeated until a specified number of iteration or a desired goal, delivering a fast evolution algorithm with parallel and noise-tolerant features. Note that the optimality is respected and evaluated in the context of the instantaneous understanding of the environment model since the reconstruction for unknown environments is a dynamic process with no a-priori information.

B. Volumetric Map-Based Navigation

Once the exploration sequence is generated, a valid motion plan is required to travel between the viewpoints. Our volumetric map-based robot navigation interface employs an external sampling-based motion planning library to search for a motion plan regarding the robot constraints. The planning problem is formatted using a compound planning space which includes the robot kinematic and dynamic constraints. The collision check is performed using the instantaneous volumetric map at the time the motion planner is engaged. The motion planner samples the continuous space and traces back the octree to check the corresponding volumetric cell occupancy. As the volumetric map is dynamically updated, the cell occupancy may differ from time to time. In order to guarantee a safe navigation, the unknown space and the occupied space are both treated as invalid, only the free space is considered valid Furthermore, a bounding box of the robot size is employed to check for the surroundings of the sample to ensure clearance. The same applies to the motion validator. Local trajectory is generated using CHOMP [18] solver for implementation on robots regarding their kinematics and dynamics.

C. Rotating Laser-Based 3-D SLAM

To fully explore the unknown environment in GPS-denied environment and reconstruct the 3-D map, a stable perception module with sufficient sensing range is essential. Besides using off-the-shelf vision sensors which may suffer from the illumination variation, we design a light-weight and compact rotating laser module with large view angle for 3-D SLAM, as shown in Fig. 1. Two configurations are exploited for experiments of mapping different environments in Section V, providing a horizontal 360° coverage and a vertical 360° coverage respectively. To cope with the sensor data distortion by such rotating mechanism on a moving UAV platform, a novel 3-D motion estimation and mapping approach is proposed for estimating the motion and mitigating the distortion. Due to space limitation, we direct the readers to the online archive of this part for greater details: https://www.dropbox.com/s/v841k5avf1ss7h3/motion_estimation.pdf?dl=0

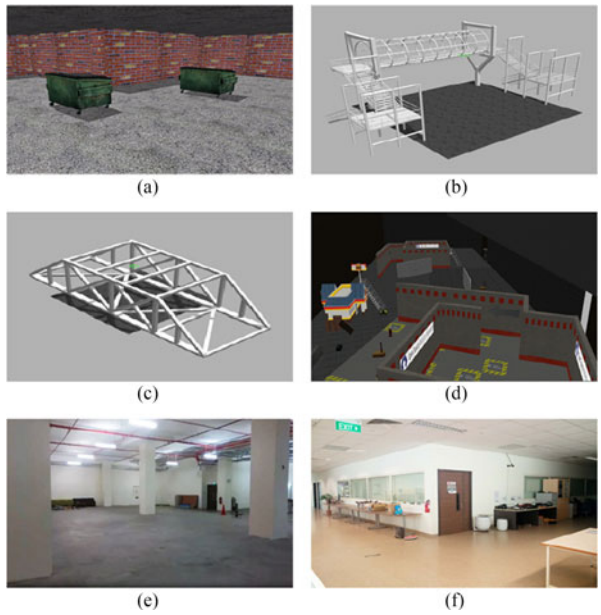

Fig. 3. Simulated and real environments for testing the framework. (a) A 18 m*18 m*4 m Chamber, (b) A 30m*30m*10m Footbridge, (c) A 50 m*20m*15 m Bridge, (d) A 90 m*35 m*8m Warehouse, (e) A UAV test arena, (f) An office corridor environment

TABLE I
PLANNER-SPECIFIC PARAMETER SETTINGS

	Receding Horizon	Receding Horizon	Ours
Environment	RRT Max Edge Length	N_{max}	η
Chamber	1m (default)	15 (default)	0.1
Footbridge	2m	15	0.05
Bridge	3m	20	0.05
Warehouse	3m	30	0.01

V. SIMULATION AND EXPERIMENTAL RESULTS

This section presents the simulation and experimental results for effective unknown environments exploration and 3-D mapping using our proposed framework with a UAV platform. Four simulated worlds (Fig. 3(a)–(d)) with increasing scales and complex structures and two real-world scenarios (Fig. 3(e) and (f)) are utilized to evaluate the performance of our system.

The simulation is conducted using the MAV Gazebo simulator RotorS as described in [10]. Stereo-vision camera as well as our rotating laser module are employed as the two main sensor types for the reconstruction work. In order to investigate the performance from both quantitative and qualitative perspectives, we conduct multiple simulation runs using the stereo-vision sensor in all simulated worlds and make comparisons on the exploration result with the state-of-the-art Receding Horizon NBV Planner in [3]. We configure our planner with fixed parameters and adopt the default settings for the Receding Horizon (RH) planner which are kept unchanged throughout the simulation runs except for several planner-specific parameters shown in Table I. The scenario-specific parameters, including the exploration volume, map resolution, robot kinematics and dynamics, sensor model, etc., are kept the same for both planners in each scenario, though they may be varied across different scenarios. Since the two planners have different auto-termination criteria

Authorized licensed use limited to: University of Tokyo. Downloaded on June 21,2025 at 07:59:21 UTC from IEEE Xplore. Restrictions apply.

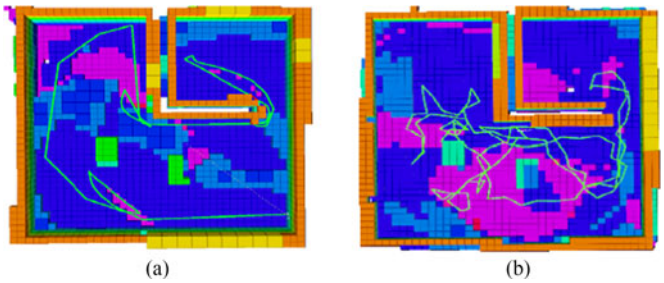

Fig. 4. Exploration results for simulated chamber using a stereo vision sensor with a FoV of 100° * 60°, and sensor range $R_s = 6\,\mathrm{m}$. Map resolution = 0.4 m. Exploration paths highlighted in 3D green line segments. (a) Ours, (b) Receding Horizon.

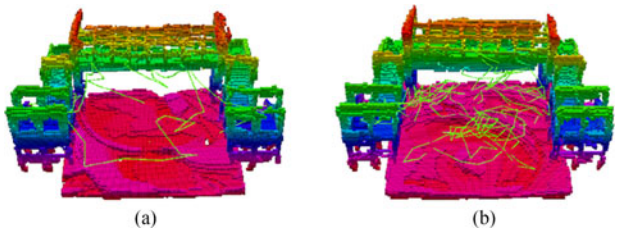

Fig. 5. Exploration results for simulated footbridge using a stereo vision sensor with a FoV of 100° * 60°, and sensor range $R_s = 10\,\mathrm{m}$. Map resolution = 0.25 m. Exploration paths highlighted in 3D green line segments. (a) Ours, (b) Receding Horizon.

which make it hard to evaluate the results on a same basis, we force the exploration termination to be triggered by either 95% of the specified bounding exploration volume been mapped or the maximum exploration is reached, with the latter being considered as unfinished tasks.

Figs. 4–7 show the exploration and mapping instances using vision sensor for the four simulated worlds respectively. Both the mapped environments and the corresponding exploration paths are displayed in the figures (the colors of Octomap correspond to height variations). We observe that our planner generally returns smoother and shorter exploration paths, while the Receding Horizon planner tends to give more complex and longer paths, which becomes more and more apparent as the complexity and scale of the environment rise up. This is a reflection of the greedy-exploration features of such planners. Its receding and greedy best-branch-guided scheme is subject to frequent changes of the best branch when there are discontinuous unknown regions nearby. The resulting plans drive the agent in back-and-forth reciprocating motions, behaving like a traveler surrounded by road branches hesitating about which direction to follow. The greedy behavior is also typical when the sensor model is constrained, which can get the agent trapped by local optimums such as "cavities" (unknown space surrounded by free space) that need numerous local observations to "fill-up". This is reflected as clusters and windings of the path generated by the Receding Horizon planner in Figs. 5–7. Such greedy behaviors overly dedicate the planning efforts to short-term local promising spaces which may possibly be covered by proper exploration and observations in the long-term global context, therefore potentially holds back the overall exploration efficiency.

It is noted that sometimes the paths generated by our planner also get affected by the sensor limitations (e.g., portions of

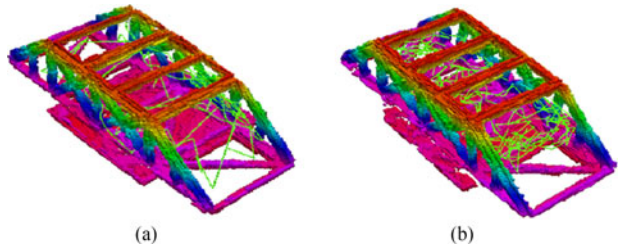

Fig. 6. Exploration results for simulated bridge using a stereo vision sensor with a FoV of 100° * 60°, and sensor range $R_s = 10\,\mathrm{m}$. Map resolution = 0.25 m. Exploration paths highlighted in 3D green line segments. (a) Ours, (b) Receding Horizon.

reciprocating motion between the left and right sections of the Warehouse scenario in Fig. 7). This is due to small FoV limiting the incremental information integration which leads to possible limited partial understanding of the contextual information in large environments. Traveling back is then needed to further explore and exploit the local information so as to perform escape from local optima and approach the global optimum. As a comparison, the rotating laser module provides a much larger coverage of surrounding local space and therefore yields smoother exploration paths in better sense of global optimality (shown in Fig. 8). Increasing sensor range from 30 to 45 m only results in a slight enhancement of the exploration path, not as prevalent as the influence of FoV. This reveals that our framework is more sensitive to sensor FoV rather than sensor range. In order to verify the global optimality of our approach, we employ the rotating laser ($R_s = 30\,\mathrm{s}$) with horizontally configured rotating laser module ($R_s = 30\,\mathrm{s}$) and conducted the simulation using the mesh of a Office Corridor environment whose optimal exploration path is empirically known (shown in Fig. 9). The sensor model employed is our . The resulting exploration instance shows a typical exploration path roughly following the groundtruth with tolerable deviation due to the stochastic nature of the sampling technique, which is considered near optimal. In the near future, we plan to develop more promising heuristics to help help ease the senor limitation and associate the global context to further approach the groundtruth optimality.

Each simulation is run 10 times to provide quantitative evaluations. In the Chamber scenario (Fig. 4), we use the default parameters of the Receding Horizon planner, which works fine within the 1500s maximum exploration time. However, the default values frequently end up with unfinished tasks in the other three environments (4000 s max exploration time for footbridge and bridge, 8000 s for Warehouse). The agent (UAV) get stuck in certain area for long periods, leaving the remaining regions unexplored. This is due to the limited tree expansion by these parameters so that the agent tends to explore nearer regions rather than further regions. Increasing these parameters helps with the situation but can also amplify the drawback of the greedy behavior, thus these parameters have to be smartly tuned according to the environment complexity and scale. We adopt values in Table I (adjusted in an increasing order to reflect the environment sizes) for the Receding Horizon planner parameters (second and third columns) in order to obtain 10 successful runs in each scenario. On the other hand, our planner does not face such limitation and appears more robust over a variety of environments. Only the depreciation factor η is varied (Table I last column) to adjust the penalization on distance to match the environment

Authorized licensed use limited to: University of Tokyo. Downloaded on June 21,2025 at 07:59:21 UTC from IEEE Xplore. Restrictions apply.

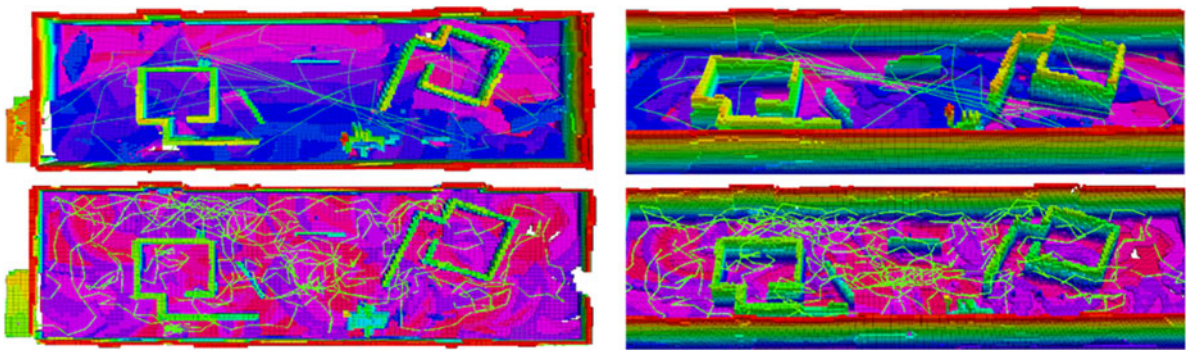

Fig. 7. Exploration results for simulated warehouse environment using a vision sensor (FoV $= 100^\circ * 60^\circ$, $R_s = 10\,\mathrm{m}$). Map resolution $= 0.4$ m. On the left shows the overview comparison (in top view) of our planner (above) and the Receding Horizon planner (below). On the right shows the corresponding zoomed 3D views for better visualization. Exploration paths are highlighted in green line segments.

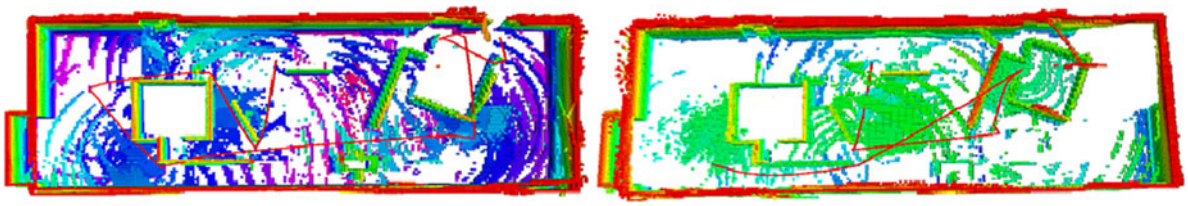

Fig. 8. Exploration results (top view) for simulated warehouse environment. On the left shows the map constructed using a rotating laser with range $R_s = 30\,\mathrm{m}$. On the right with increased sensor range $R_s = 45\,\mathrm{m}$. Exploration paths highlighted in red.

Fig. 9. Exploration result (top view) of a simulated office environment using the horizontal configuration of rotating laser module with range $R_s = 30\,\mathrm{m}$. The executed exploration path is highlighted in red line segments. The groundtruth optimal exploration path in dashed black line segments.

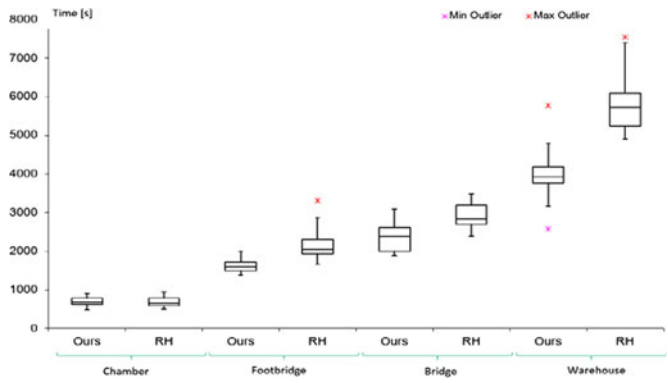

Fig. 10. Statistics of 10 successful runs for the 4 simulated environments using our approach and the Receding Horizon (RH) planner.

scale. Statistical results in Fig. 10 demonstrate the overall performances of both planners. For the simplest Chamber scenario, the performances seem comparable with the median total exploration time (our planner vs. the Receding Horizon planner) being 685.07 vs. 657.83 s and the corresponding IQR being 146.65 vs. 198.20 s. As the complexity and scale increase, the advantage of our planner becomes more prevalent with the median values being 1595.74 vs. 2042.42 s, 2383.58 vs. 2832.27 s, 3933.30 vs. 5729.05 s, and the IQR values being 226.82 vs. 384.39 s, 608.46 vs. 483.01 s, 410.07 vs. 866.29 s for the other three scenarios respectively.

The results for the real-world explorations are displayed in Fig. 11. The rotating laser module is configured respectively as horizontal and vertical for the UAV Test Area (a) and the Office Corridor (b). Since the vertical configuration covers 360° in the vertical plane, the map is displayed in a closed form to demonstrate this feature. The corresponding exploration path is highlighted in the point clouds visualization. In both cases, the framework show complete reconstruction of the environment structure with a reasonable exploration path of short distance, demonstrating the practical feasibility of our framework.

The above evaluations verify the competitiveness of our approach, especially for complex and large scale environments exploration and reconstruction. Although the presence of sensor

Authorized licensed use limited to: University of Tokyo. Downloaded on June 21,2025 at 07:59:21 UTC from IEEE Xplore. Restrictions apply.

(a) (b)

Fig. 11. Exploration results (top view) for real environments using our planner and the rotating laser module with range $R_s = 30\,\mathrm{m}$. (a) shows the map and exploration path of the UAV test area using the horizontal configuration. (b) shows the path with point clouds (left) and the corresponding map (right) of the office corridor environment, portions of scattering due to glass doors. Exploration paths are highlighted in red line segments.

limitations can sometimes affect the quality of path, our planner generally demonstrates reasonably improved exploration efficiency in terms of overall path simplicity and total exploration time.

VI. Conclusion

In this paper, we present an optimized next view planning framework for autonomous 3-D unknown environment exploration and reconstruction without a-priori knowledge. An information gain-based 2-stage planning algorithm is implemented iteratively to perform full coverage sampling of the frontier boundary region and to subsequently solve for FSOTSP instances. Moreover, a volumetric map-based navigation interface and a rotating laser-based SLAM module are exploited to generate valid motion plans and construct the environment model, respectively. We use simulation scenarios and experimental runs with our UAV platform to demonstrate the effectiveness of the proposed framework. The results show the UAV efficiently performs full-coverage exploration and structural model reconstruction with improved overall exploration efficiency.

Acknowledgment

The authors would also like to thank the Department of Mechanical Engineering for resources and workshops offered on related topics.

References

[1] M. Abdelguerfi, *3D Synthetic Environment Reconstruction*. New York, NY, USA: Springer, 2013.

[2] A. Bircher *et al.*, "Structural inspection path planning via iterative viewpoint resampling with application to aerial robotics," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 6423–6430.

[3] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart, "Receding horizon next-best-view planner for 3d exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 1462–1468.

[4] P. S. Blaer and P. K. Allen, "Data acquisition and view planning for 3-D modeling tasks," *IROS. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, pp. 417–422, 2007.

[5] D. Borrmann *et al.*, "The project thermalmapper–thermal 3D mapping of indoor environments for saving energy," *IFAC Proc. Vol.*, vol. 45, no. 22, pp. 31–38, 2012.

[6] C. Cadena *et al.*, "Simultaneous localization and mapping: Present, future, and the robust-perception age," arXiv preprint arXiv:1606.05830, 2016.

[7] M. Dakulovic, S. Iles, and I. Petrovic, "Exploration and mapping of unknown polygonal environments based on uncertain range data," *Autom.–J. Control, Meas., Electron., Comput. Commun.*, vol. 52, no. 2, 2011.

[8] H. Durrant-Whyte and T. Bailey, "Simultaneous localisation and mapping (slam): Part i the essential algorithms," *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 99–110, Jun. 2006.

[9] A. Ekman, A. Torne, and D. Stromberg, "Exploration of polygonal environments using range data," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 27, no. 2, pp. 250–255, Apr. 1997.

[10] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, "Rotors—A modular gazebo mav simulator framework," *Robot Operating System (ROS):*, Springer, vol. 1, pp. 595–625, 2016.

[11] H. González-Banos, "A randomized art-gallery algorithm for sensor placement," in *Proc. 17th Annu. Symp. Comput. Geom.*, 2001, pp. 232–240.

[12] S. Isler, R. Sabzevari, J. Delmerico, and D. Scaramuzza, "An information gain formulation for active volumetric 3d reconstruction," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 3477–3484.

[13] K. Lenac, A. Kitanov, I. Maurović, M. Dakulović, and I. Petrović, *Fast Active SLAM for Accurate and Complete Coverage Mapping of Unknown Environments*. New York, NY, USA: Springer, 2016, pp. 415–428.

[14] R. A. Martin, I. Rojas, K. Franke, and J. D. Hedengren, "Evolutionary view planning for optimized UAV terrain modeling in a simulated environment," *Remote Sens.*, vol. 8, no. 1, p. 26, 2016.

[15] M. Montemerlo and S. Thrun, *FastSLAM: A Scalable Method for the Simultaneous Localization and Mapping Problem in Robotics* (Springer Tracts in Advanced Robotics). Secaucus, NJ, USA: Springer-Verlag, 2007.

[16] Y. Okada and J. Miura, "Exploration and observation planning for 3d indoor mapping," in *Proc. IEEE/SICE Int. Symp. Syst. Integr.*, 2015, pp. 599–604.

[17] C. Potthast and G. S. Sukhatme, "A probabilistic framework for next best view estimation in a cluttered environment," *J. Vis. Commun. Image Represent.*, vol. 25, no. 1, pp. 148–164, 2014.

[18] N. Ratliff, M. Zucker, J. A. Bagnell, and S. Srinivasa, "CHOMP: Gradient optimization techniques for efficient motion planning," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 489–494.

[19] S. Singh and E. A. Lodhi, "Study of variation in TSP using genetic algorithm and its operator comparison," *Int. J. Soft Comput. Eng.*, vol. 3, no. 2, pp. 264–267, 2013.

[20] J. I. Vasquez-Gomez, L. E. Sucar, R. Murrieta-Cid, and E. Lopez-Damian, "Volumetric next-best-view planning for 3D object reconstruction with positioning error," *Int. J. Adv Robot. Syst.*, vol. 11, no. 10, p. 159, 2014.

[21] J. W. Weingarten and R. Siegwart, "EKF-based 3d SLAM for structured environment reconstruction," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2005, pp. 3834–3839.

[22] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *Proc. IEEE Int. Symp. Comput. Intell. Robot. Autom. CIRA'97., Proc., 1997 IEEE Int. Symp.*, 1997, pp. 146–151.

[23] L. Yoder and S. Scherer, "Autonomous exploration for infrastructure modeling with a micro aerial vehicle," in *Proc. Field Serv. Robot.*, 2016, pp. 427–440.

Authorized licensed use limited to: University of Tokyo. Downloaded on June 21,2025 at 07:59:21 UTC from IEEE Xplore. Restrictions apply.