Fast and Compute-Efficient Sampling-Based Local Exploration Planning via Distribution Learning

Lukas Schmid, Graduate Student Member, IEEE, Chao Ni, Yuliang Zhong, Roland Siegwart, Fellow, IEEE, and Olov Andersson, Member, IEEE

Abstract—Exploration is a fundamental problem in robotics. While sampling-based planners have shown high performance and robustness, they are oftentimes compute intensive and can exhibit high variance. To this end, we propose to learn both components of sampling-based exploration. We present a method to directly learn an underlying informed distribution of views based on the spatial context in the robot’s map, and further explore a variety of methods to also learn the information gain of each sample. We show in thorough experimental evaluation that our proposed system improves exploration performance by up to 28% over classical methods, and find that learning the gains in addition to the sampling distribution can provide favorable performance vs. compute trade-offs for compute-constrained systems. We demonstrate in simulation and on a low-cost mobile robot that our system generalizes well to varying environments.

Index Terms—Reactive and sensor-based planning, integrated planning and learning, motion and path planning.

I. INTRODUCTION

EXPLORATION of unknown environments is a fundamental problem in robotics. It is essential for a variety of applications, ranging from consumer robotics to autonomous surveying and inspection, as well as search and rescue. In this paper, we consider the problem of exploring unknown environments represented by standard occupancy maps, encoding free, occupied and unknown space. This common map representation can be obtained from any volumetric mapping system [1]–[3] and enables a ground robot to plan in continuous 2D space.

In recent years, a clear trend has emerged to split the exploration problem into the distinct sub-problems of local and global exploration [4]–[8]. This can be explained by the conflicting objectives of complete coverage and fast exploration progress, which can be tackled independently in local and global planning. In local exploration, the goal is to generate a safe path that uncovers as much of yet unmapped space in as little time as possible. Complementary, global exploration aims at identifying a traversal order of frontiers, the boundaries between known and unknown space, to escape local minima and achieve complete coverage. In this work, we focus solely on the problem of local exploration.

In similar fashion, two main families of approaches, being frontier and sampling-based methods, have emerged. In frontier-based exploration [9], [10] the frontiers are directly used to compute exploration goals. In the other, sampling-based methods first, i) sample viewpoints in the mapped space, and ii) assign a utility value to views, to identify the Next-Best-View (NBV) [4]–[8], [11]–[14].

Although there exist extensions of frontier-based methods for rapid flight [10], sampling-based planning has found significantly more success in local exploration [4]–[8], [11]–[13]. This is primarily due to their ability to capture volumetric information for each view point, the possibility to account for varying information gain objectives [12], [14], and trade-off gains and costs for each viewpoint [4]–[8], [11]–[14].

However, traditional sampling-based methods also have notable shortcomings. First, they are oftentimes highly compute intensive, potentially reducing performance on compute constrained mobile robots, and leaving less capacity for other tasks such as mapping or scene understanding. Second, the performance of such methods is strongly governed by the number of samples and the randomness inherent to the sampling procedure. Therefore, many samples may be needed to identify good paths, such as e.g. when most of the space in a room is already explored, or the robot has to traverse a narrow passage.

Simultaneously, learning-based methods have shown strong potential in robotics. Typically, active planning problems are
tackled via Reinforcement Learning (RL) [15]–[19] or imitation learning [20]–[22], learning a mapping of the current state to an action. However, RL methods are known to be hard to train [16] and typically require significant simplifications.

To overcome these limitations, we propose a learning-based approach that follows the sampling-based exploration paradigm. To this end, we propose to learn an underlying informed distribution of valuable viewpoint, accounting for information gain, kinematic feasibility, collision avoidance, and dynamics-based motion cost, directly from the robot map. We further explore different variations of learning to efficiently estimate the information gain of a sample, reducing the computational requirements. We show in thorough experimental evaluation in simulated and real world experiments that this approach is interpretable and robust, simultaneously improving exploration performance while reducing computational cost, generalizing to different environments.

We make the following contributions:

- We propose a new approach to local exploration by learning the key components of sampling-based exploration planning.
- We propose a method based on conditional variational autoencoders (CVAE) to directly learn informed multi-modal sampling distributions for exploration planning from standard occupancy maps. We show that this approach is interpretable and achieves superior performance compared to conventional exploration planners and an imitation learning baseline.
- We further explore a variety of approaches to learn efficient estimates of the expensive gain computation, and find that CVAE-sampling combined with learned gains offers favorable compute vs. performance trade-offs for compute constrained systems.
- We thoroughly validate the generalization capability of our approach with both simulated and real-world experiments. Our approach transfers to a small mobile robot with low-cost sensors and maintains good performance without tuning or retraining. We make our code available as open-source.\(^1\)

II. RELATED WORK

A. Sampling-Based Local Exploration Planning

Sampling-based planning has found distinct success in exploration planning. To find the NBV, typically viewpoints are sampled, information gains and final utilities of each sample are computed, and the best view is selected. Bircher et al. [11] store samples in a Rapidly-exploring Random Tree (RRT) structure to create a sequence of traversable poses. The gain of each viewpoint is the number of observable voxels, discounted by an exponential distance term. This method has been widely used and extended in subsequent works [4]–[7]. However, these mostly innovate on the global planner and use similar local planners to [11]. An alternative to distance-based discounts is to normalize the gain by the cost [8], [12], which is naturally closer to the final exploration objective. While [12] uses this utility to create informative long-term plans for different applications, [8] shows that, for exploration, the majority of the relevant data lies within a very short planning horizon and superior performance can be achieved via local planning. All of the above methods use uniform random sampling. This is motivated by the advantage of uniform sampling being asymptotically complete, and it typically achieves good coverage in practice.

A different line of approaches attempt to manually engineer heuristics for informed sampling in exploration [13], [14], [23], based on computing the location of frontiers and sampling points around these as a heuristic for where informative poses may lie. However, this can lead to poses that are not easily accessible and come at an increased computational cost. As engineering informed and computationally cheap heuristics is generally a non-trivial problem, we argue that it can instead be learned automatically, where the full contextual information in the map can be used to inform NBV sampling.

B. Learning for Path Planning and Exploration

With recent successes of machine learning across multiple domains, leveraging learning approaches for motion planning problems has also been receiving increasing attention. However, most works so far consider the classical motion planning problem with a known map and a given goal [24]–[26]. Iccher et al. [27] also learns a sampling distribution, but only considers classical motion planning with known maps and uses a high-level representation of the environment (e.g. object positions) instead of standard occupancy maps. Some works relax the assumption of a known map but they still focus on finding a path to a given goal [28], [29].

The problem of learning to explore unknown environments has received less attention. RL approaches are in theory the most general, but as they can be difficult to train [16], these tend to rely on significant simplifications. Zhu et al. [17] consider the problem of exploring office environments and use an actor-critic RL approach to act as a filter for a classical planner, selecting one of six sectors to sample from. Similarly, Niroui et al. [18] first extract frontiers and use an RL approach to select between them. Chen et al. [19] used RL with a pre-defined quasi-random distribution of actions for a robot with omnidirectional sensor and kinematics.

Another learning approach popular for planning and control tasks is to use supervised learning to imitate a conventional planner, which can be computationally cheaper [20]. In the area of exploration planning, Reinhart et al. [21] learn to imitate a planner tailored to tunnel environments, where the action space is limited to trajectories in eight regions with a fixed length of 2 m. Englert et al. [22] similarly discretize the action space into 36 angles on a circle around the robot and show a reduction in computational cost over exhaustively evaluating the gains for all such actions. While learning holds great potential, current approaches do not allow sampling viewpoints freely in space, instead relying on pre-defined action distributions and often holonomic robots with omnidirectional sensors. Such assumptions do not always well reflect the constraints of real robots. We thus propose to combine the advantages of learning and sampling-based planners, learning both an informed sampling distribution from the robot map, as well as the information gain of each sample, to efficiently compute the utility and NBV.

III. PROBLEM STATEMENT

In this paper, we address the problem of using learning to improve sampling-based local exploration using conventional occupancy maps, visualized in Fig. 2.
Given is an area of interest represented as a voxel grid \( V \subseteq \mathbb{R}^2 \). Each voxel \( v \in V \) has a state in the robot map at time \( t \) \( M_t(v) : \mathbb{R}^2 \rightarrow \{o, f, u\} \), denoting occupied, free, and unknown areas. Initially \( M_0(v) = m_u, \forall v \in V \). Here \( V_o \subseteq V \) denotes the observable space for a given robot.

**B. NBV Planning**

At each time step \( t \), the robot is given a local map of the environment \( L_t \subseteq V \). The task is then to identify the NBV pose \( P_t = \{x_t, y_t, \theta_t\} \in \mathbb{R}^2 \times SO(2) \) within \( L_t \), where the robot moves to.

**C. Objective Function**

The objective of a planning mission can be formulated as a constrained optimization problem:

\[
P_0^*, \ldots, P_N^* = \min_{P_0, \ldots, P_N} \sum_{i=0}^{N} T(P_i) + \gamma C(P_i)
\]

subject to \( M_{N^t}(v) \neq m_u, \forall v \in V_o \).

When the planner reaches a local minimum \( \hat{P}_v \in L_t \mid M_t(v) = m_u \land \text{reachable}(v) \), i.e. there is no newly observable voxel left in the local robot map, or the planner has not observed any voxel in 5 consecutive actions, the global planner is called. We use a frontier-based global planner similar to [8] to move the robot to the closest unexplored site and continue local planning.

**A. Learning Local Sampling Distributions**

The central idea of our approach is based upon the observations, that i) in exploration planning the distribution of informative viewpoints is typically multi-modal and ii) the robot map can provide significant spatial context. We thus propose to directly learn an informed distribution, accounting for information gain, kinematic feasibility, collision avoidance, and dynamics-based motion cost, for an environment \( L_t \). Since the set of possible poses \( P_t = \{x_t, y_t, \theta_t\} \in L_t \) quickly becomes intractable to optimize over, we take advantage of sampling-based planning to generate a finite number of high utility viewpoint points, detailed in Section IV-C. We then use these poses as a supervision signal to learn an underlying informed distribution. In particular, we leverage Conditional Variational Auto-Encoders (CVAE) to capture a hidden latent distribution [30] based on training samples. An overview of our architecture is shown in Fig. 4.

1) Preliminaries: We denote \( X \) the target variable, parametrized as pose samples \( P \in \mathbb{R}^2 \times SO(2), X = \{P_i\}_{i=0}^{N} \). We denote the conditioning variable as \( Y \) and the latent representation as \( Z \). As the true distribution \( p(Z|X,Y) \) is unknown, it is approximated by a decoder \( Q_\phi(Z|X,Y) \), where \( Q \) is parametrized as a neural network with parameters \( \phi \). Similarly, the generative distribution \( P_\psi(X|Z,Y) \) is modelled as the decoder network with parameters \( \psi \). We model the hidden latent distribution \( p(Z|Y) \) as Gaussian, and use the reparametrization trick to propagate gradients through the sampling step of \( Z \). The non-linearity of the encoder \( P_\psi \) is intractable, we optimize the variational lower bound \( \mathcal{L} \leq \log p(X|Y) \):

\[
\mathcal{L} = \mathbb{E}_{Z \sim Q_\phi}[ \log p_\psi(X|Z,Y) ] - D_{KL}(Q_\phi(Z|X,Y)||p(Z|Y))
\]

where \( D_{KL}() \) is the Kullback-Leibler divergence. \( \mathcal{L} \) can be approximated by Monte Carlo methods and efficiently optimized by standard learning techniques [31], [32].

2) Map Encoding: Since the robot environment is represented by a high dimensional occupancy map \( L_t \in \mathbb{R}^{N_L \times N_V} \), we encode it into a more compact conditioning \( Y \in \mathbb{R}^{N_Y} \). Although numerous methods are permissible, we found that pooling can achieve highly salient compression with very little compute cost. We thus represent the map encoder as a single \( N_{pool} \times N_{pool} \) max-pooling layer, with \( m_u > m_f > m_f \) and \( N_{pool} = N_L \sqrt{N_V} \). \( Y \) is further one-hot-encoded to capture the discrete nature of \( V \). We use a conditioning size \( N_Y = 100 \), and allow for arbitrary map resolutions \( N_V \). In our experiments \( N_L = 50 \). For efficient array slicing, we extract \( L_t \) in world orientation and additionally append the robot angle \( \theta_t \) w.r.t. \( L_t \) to \( Y \).

3) Encoder and Decoder: The encoder \( Q_\phi \) and decoder \( P_\psi \) are parametrized as Multi-Layer Perceptrons (MLPs), with 4 layers of 512 hidden units each, using dropout for regularization. They map to a latent representation \( z \in \mathbb{R}^4 \). \( Q_\phi \) and \( P_\psi \) are then trained jointly to optimize the loss.
4) **Loss:** We use $\mathcal{L}$ of (3) as the loss function. The first term of $\mathcal{L}$ represents the ability of $P_\phi$ to reconstruct the input distribution $X$, which can be modeled as

$$L_{rec} = \frac{1}{N_x} \sum_{i=0}^{i=N_x} \|P_i - P'_i\|^2$$  \hspace{1cm} (4)

where we use the Euclidean norm for $\|\{x_i, y_i\} - \{x'_i, y'_i\}\|$ and the shortest arc for $\|\theta_i - \theta'_i\|$.

The second term of $\mathcal{L}$ encourages the input encoding $Z \sim Q_\theta$ to be close to the prior distribution $p(Z|Y) = \mathcal{N}(0, I)$. To trade off these objectives, the second term is often weighted by $\lambda_{r eg}$, but we empirically found 1 to work well.

5) **Inference:** At test time, it is sufficient to simply draw a latent seed $Z \sim \mathcal{N}(0, I)$ from the prior distribution, and feed it to the decoder with the map conditioning $Y$ to obtain a learned sample.

**B. Learning Sample Information Gains**

The second important component in sampling-based planning is to identify the highest utility sample. We adopt the formulation of [12] and differentiate between total utility $u(P)$, trading off gain $g(P)$ and cost $c(P)$ of a sample $P$. We employ the common exploration gain formulation:

$$g(P) = \sum_{v \in \text{Visible}(P)} f(v) \text{, } f(v) = \begin{cases} 1 & \text{if } M(v) = m_u \\ 0 & \text{else} \end{cases}$$  \hspace{1cm} (5)

and use the estimated traversal time $\hat{T}(P)$ as cost $c(P)$. Differentiating gain and cost in this formulation allows our method to be applied to arbitrary cost functions, e.g. for different robot and motion models, without retraining. The final utility to select the NBV is computed as $u(P) = g(P) / c(P)$ [8], [12], NBV = $\arg \max_c \{u(P_i)\}$. Typically, $g(P)$ is computed via ray-casting.

We term this method CVAE. However, since $g(P)$ is oftentimes expensive to compute [12], we propose to directly learn it. For this, we explore three methods.

An intuitive approach is to jointly estimate $P'$ and $g(P')$. We thus directly add $g$ as a coupled target variable to the CVAE data space, i.e. $X = \{P_i\}_{i=0, \ldots, N_X}, P'_i = \{x_i, y_i, \theta_i, g_i\}$. We refer to this joint method as CVAE+$GJ$. However, this approach has the potential limitations of degrading the sampling quality to capture the gain. Therefore, we also train an independent network $G_\xi(P, Y)$ where $G_\xi$ is an MLP of identical architecture as $Q_\theta$, with parameters $\xi$. To train $G_\xi$, we also add negative, i.e. random, samples to the training set of Section IV-C to not bias the estimator. Using this approach, $G_\xi$ can be combined with any sampling method, referred to as CVAE+$G$ and Uniform+$G$.

Third, while we found the simple map encoding to work well for the CVAE, for the gain estimation we also present more complex map encoder components, parametrized as a Convolutional Neural Networks (CNN) consisting of three $5 \times 5$ convolutions and a max-pooling layer. These have higher compute cost than the simple encoder, but can result in more complex features. The map encoder is jointly trained with the gain-estimator. We compare this when paired with both the CVAE, and conventional uniform sampling, methods denoted CVAE+$GCNN$ and Uniform+$GCNN$, respectively.

**C. Dataset Generation**

To train our networks, a dataset is collected. To achieve a diverse set of environments, simulated worlds are procedurally generated. We implement two types of environments, a classical Maze for indoor scenes and Cluttered with a lot of fuzzy surfaces, shown in Fig. 5. The simulator and world generation are made open source with the code¹. Note that only worlds of type Maze are used for training. We use the Uniform planner of Section V to explore each world. For each seen local map, we generate a set of high utility samples $X$ by running the Uniform planner.
with $N = 25$ samples $N_X = 20$ times, storing only the NBVs. Examples of this are given in Fig. 6 (top row). To optimize $\theta_i$, we further employ orientation optimization [4], [7], [12] of each pose. The total dataset $D = \{L_i, X_i\}_{i=0,...,N_D}$ contains $N_D = 200k$ samples from 10k worlds, 160k for validation.

First, we perform thorough experimental evaluation of all methods in simulation. All methods are run on a separate test set consisting of 5 worlds. For comparability, all methods are run 3 times from identical start poses. The mean and standard deviation over these 15 runs is reported. An omni-directional robot with linear and angular velocities of $v_{\text{max}} = 1$ m/s and $\omega_{\text{max}} = 1$ rad/s, equipped with an RGB-D camera with FoV $= 90^\circ$ and range $r = 5$ m, is modelled. We use a local map of twice the sensing range as suggested in [8].

We compare our proposed methods against the following set of baselines. Uniform is a classical sampling-based planner, that uses uniform random sampling to generate $N$ reachable poses $P_i$, computes $g(P_i)$ via ray-casting, and chooses $P_t = \text{arg max}_i u(P_i)$. We further compare against the established method of [11], termed RH-NBVP. As a learning-based method, Imitation directly predicts the best $P_t$ from the map conditioning on $Y$. This is a conventional behavior cloning approach using regression on the targets with square loss [22]. To this end, we trained a separate network $R_{\phi}(Y)$ with identical architecture to $Q_{\phi}$ on our dataset. We also tried to train an RL baseline based on GA3C [15] like in [17]. However, RL is known to be difficult to train [16] and after considerable effort to tune parameters, architecture, and rewards, no presentable results were achieved. While RL holds great promise, it appears our CVAE-based approach is easier to train, at least on our problem with more complex robot sensor and motion models than in related work on RL.

### V. EXPERIMENTAL SETUP

#### A. Simulation

First, we qualitatively analyze the performance of the proposed distribution learning method, shown in Fig. 6. The wide spread of NBVs highlights the high variance and multi-modality in sampling-based exploration planning. We observe that our method is well able to capture this multi-modality and produces highly informative samples. This is also reflected in Table I showing the utility of chosen actions. Our method produces actions close to the training data, showing a fivefold improvement over Uniform for low $N$. As $N$ and therefore coverage increases, the choice of gain estimator becomes more important.

#### B. Baselines

With $N = 25$ samples $N_X = 20$ times, storing only the NBVs. Examples of this are given in Fig. 6 (top row). To optimize $\theta_i$, we further employ orientation optimization [4], [7], [12] of each pose. The total dataset $D = \{L_i, X_i\}_{i=0,...,N_D}$ contains $N_D = 200k$ samples from 10k worlds, 160k for training and 40k for validation.

### VI. RESULTS

#### A. Distribution and Gain Learning

First, we qualitatively analyze the performance of the proposed distribution learning method, shown in Fig. 6. The wide spread of NBVs highlights the high variance and multi-modality in sampling-based exploration planning. We observe that our method is well able to capture this multi-modality and produces highly informative samples. This is also reflected in Table I showing the utility of chosen actions. Our method produces actions close to the training data, showing a fivefold improvement over Uniform for low $N$. As $N$ and therefore coverage increases, the choice of gain estimator becomes more important.

#### B. Exploration Performance

To evaluate exploration performance, the exploration progress for each method, using $N = \{1, 5, 10, 20, 50\}$ samples per planning step, is shown in Fig. 7. The time and distance until exploration is complete is shown in Table II.

For $N = 1$ the importance of the learnt sampling distribution is evident. By drawing a single sample from the model, the
CVAE-based methods show significantly higher performance than their classical counterparts. As the number of samples increases, CVAE further improves in performance, reducing the time until exploration is completed by 42% at 50 samples. Naturally, as the coverage becomes more complete for increasing $N$, the influence of the sampling distribution decreases. Notably, even for 50 samples, after which no significant performance improvements were found, CVAE still shows similar or better performance than uniform. This suggests that the optimal viewpoint is contained within the learned distribution.

The importance of the gain computation becomes evident as the number of samples increases. Here, the ray-casting based methods act as the upper bound since they compute the true gain, reflected in the high performance of CVAE and Uniform. Importantly, methods that learn an information gain improve with more samples being drawn, with the exception of CVAE+G. This is explained by the high variance of the $\max(\cdot)$ operator, where a single noisy or overfitted prediction can decrease exploration performance, as is also observed in Table I. In contrast to this, the CNN-based gain estimator is able to better predict the information gains, leading to improved performance for more samples with CVAE+GCNN. The trend is even more pronounced in Uniform+GCNN. This behavior for CVAE+G and CVAE+GCNN highlights the interplay between sampling and gain evaluation and opens interesting avenues for future research. The joint architecture in CVAE+GJ has to trade off learning informed sampling and accurate gains at the same time and is not able to improve performance. RH-NBVP exhibits slightly lower performance than the other methods. This can be explained by the sub-optimal wiring of its RRT, whereas the other methods including Uniform follow the approach of [8] and plan for a short horizon to incorporate the new information.

We observe that the distribution learning methods improve with more samples being added, compared to imitation learning. This suggests that the multi-modal nature of sampling translates to increased exploration performance. In addition, Imitation is the only method that failed to produce a feasible path for 2.5% cases.

The exploration time is primarily governed by the quality of the selected view points, which also manifests in the travel distances shown in Table II. The only exception is RH-NBVP, which does not account for the robot’s dynamics and chooses shorter but slower paths.

C. Generalization

To evaluate the generalization performance of our method, we conduct experiments in Cluttered environments. Note that all models were only trained on Maze data, which is distinctly different as shown in Fig. 5. The results for $N = \{5, 50\}$ are presented in Fig. 8.

Even in this novel environment, we find that the performance is highly similar to the Maze environment, suggesting our method generalizes well to other environments.

D. Computation Cost

Computing power is a crucial and limited resource on a mobile robot. Depending on hardware limitations, energy constraints, and the other tasks the robot has to perform, different weights of the computation objective $\gamma$ (Section III) are desirable. This compute vs. performance trade-off for all methods is presented as a Pareto curve in Fig. 9.
by up to 28% at similar cost, or achieving similar performance while reducing computation 5.3 times.

The advantage of learning the information gain is apparent in the significantly reduced compute cost. CVAE+GCNN exhibits the best performance in the low-compute area, although the performance gain for more samples is less pronounced. Similar behavior is also apparent for random sampling. While Uniform+G/+GCNN benefit from more samples, they only offer improved performance in the low-cost regime. As expected, Imitation offers a very fast but rather poor result.

E. Robot Experiments

To further validate our approach, experiments on a low-cost mobile robot are conducted. We employ a Turtlebot equipped with a realsense D435 depth camera, shown in Fig. 12, to explore a challenging, cluttered indoor environment, shown in Fig. 1. Three experiments of CVAE and Uniform for $N=10$ are performed.

Fig. 10 shows example local maps, drawn samples, and selected NBVs from the real system. Due to the high sensor noise and localization errors, the map extracted from [3] after depth fusion is notably more noisy than in simulation. Together with the different scale of the narrow environment, this constitutes a major domain shift. Nonetheless, our network purely trained in simulation is able to generate meaningful samples most of the time. We observe that the Uniform planner exhibits high variance, sometimes generating close to optimal views, at other times missing the obvious.

This is also reflected in the quantitative results reported in Fig. 11. While both methods are able to successfully explore the whole environment, CVAE tends to show quicker and more consistent exploration. A final map obtained after exploration is shown in Fig. 11 bottom. While the map quality primarily depends on the localization and sensing noise, we observe that our planner thoroughly explores all free space. Lastly, Fig. 12, right, shows the computation time per planning step. CVAE shows improved computation cost, as many more random samples need to be checked to achieve 10 feasible poses in this complex environment.

VII. CONCLUSION

In this work, we consider the problem of learning sampling-based local exploration planning. We propose a CVAE-based
approach to directly predict an, often multi-modal, informed sample distribution directly from standard partially-observed occupancy maps. We show in thorough experimental evaluation that the presented approach can either improve exploration performance by up to 28% or match performance and reduce computation by a factor of five compared to conventional approaches. Furthermore, we explore a variety of methods to combine sampling and gain computation. We find that learning the information gain with a CNN map encoder, in combination with the CV AE-based sampling strategy, leads to favorable performance vs. compute trade-offs for compute constrained systems. We show both in simulation and on a low-cost mobile robot that our method generalizes well to different environments.

REFERENCES

[1] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, “OctoMap: An efficient probabilistic 3D mapping framework based on octrees,” Auton. Robots, vol. 34, no. 3, pp. 189–206, 2013.
[2] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, “Voxblox: Incremental 3D euclidean signed distance fields for on-board MAV planning,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., IROS, 2017, pp. 1366–1373.
[3] L. Schmid et al., “Panoptic multi-TSDFs: A flexible representation for online multi-resolution volumetric mapping and long-term dynamic scene consistency,” 2021, arXiv:2109.10365.
[4] C. Witting, M. Fehr, R. Bähnemann, H. Oleynikova, and R. Siegwart, “History-aware autonomous exploration in confined environments using MAVs,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., IROS, 2018, pp. 1–9.
[5] Z. Meng et al., “A two-stage optimized next-view planning framework for 3-D unknown environment exploration, and structural reconstruction,” IEEE Robot. Automat. Lett., vol. 2, no. 3, pp. 1680–1687, Jul. 2017.
[6] T. Dang, S. Khattak, F. Mascari, and K. Alexis, “Explore locally, plan globally: A path planning framework for autonomous robotic exploration in subterranean environments,” in Proc. IEEE 19th Int. Conf. Adv. Robot., ICAR, 2019, pp. 9–16.
[7] M. Selin, M. Tiger, D. Duberg, F. Heinzt, and P. Jensfelt, “Efficient autonomous exploration planning of large-scale 3-D environments,” IEEE Robot. Automat. Lett., vol. 4, no. 2, pp. 1699–1706, Apr. 2019.
[8] L. Schmid, V. Reijgwart, L. Ott, J. Nieto, R. Siegwart, and C. Cadena, “A unified approach for autonomous volumetric exploration of large scale environments under severe odometry drift,” IEEE Robot. Automat. Lett., vol. 6, no. 3, pp. 4504–4511, Jul. 2021.
[9] B. Yamauchi, “A frontier-based approach for autonomous exploration,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., IROS, 2017, pp. 2135–2142.
[10] T. Cieslewski, E. Kaufmann, and D. Scaramuzza, “Rapid exploration with multi-rotors: A frontier selection method for high speed flight,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., IROS, 2017, pp. 2135–2142.
[11] 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. Automat., ICRA, 2016, pp. 1462–1468.
[12] L. Schmid, M. Pantic, R. Khanna, L. Ott, R. Siegwart, and J. Nieto, “An efficient sampling-based method for online informative path planning in unknown environments,” IEEE Robot. Automat. Lett., vol. 5, no. 2, pp. 1500–1507, Apr. 2020.