Auton Robot (2018) 42:257–272
https://doi.org/10.1007/s10514-017-9669-2

Environment model adaptation for mobile robot exploration

Erik Nelson[1] · Micah Corah[2] · Nathan Michael[2]

Received: 29 February 2016 / Accepted: 17 October 2017 / Published online: 8 November 2017
© Springer Science+Business Media, LLC 2017

Abstract In this work, we propose a methodology to adapt a mobile robot's environment model during exploration as a means of decreasing the computational complexity associated with information metric evaluation and consequently increasing the speed at which the system is able to plan actions and travel through an unknown region given finite computational resources. Recent advances in exploration compute control actions by optimizing information-theoretic metrics on the robot's map. These metrics are generally computationally expensive to evaluate, limiting the speed at which a robot is able to explore. To reduce computational cost, we propose keeping two representations of the environment: one full resolution representation for planning and collision checking, and another with a coarse resolution for rapidly evaluating the informativeness of planned actions. To generate the coarse representation, we employ the Principal of Relevant Information from rate distortion theory to compress a robot's occupancy grid map. We then propose a method for selecting a coarse representation that sacrifices a minimal amount of information about expected future sensor measurements using the Information Bottleneck Method. We outline an adaptive strategy that changes the robot's environment representation in response to its surroundings to maximize the computational efficiency of exploration. On computationally constrained systems, this reduction in complexity enables planning over longer predictive horizons, leading to faster navigation. We simulate and experimentally evaluate mutual information based exploration through cluttered indoor environments with exploration rates that adapt based on environment complexity leading to an order-of-magnitude increase in the maximum rate of exploration in contrast to non-adaptive techniques given the same finite computational resources.

Keywords Active perception · Mobile robot exploration · Sensor based planning · Mapping

1 Introduction

Robots are emerging from controlled factories and laboratories into our homes, workplaces, roads, and airspace. Alongside their transition into these unstructured and transient environments comes their need to be able to explore, characterizing and cataloging their surroundings as they navigate. Recent robotic exploration methods plan actions that reduce map uncertainty by optimizing information-theoretic metrics (Bourgault et al. 2002; Amigoni and Caglioti 2010; Julian et al. 2013; Julian 2013). While these techniques yield increased exploration performance (Charrow et al. 2015b) in contrast to geometric techniques that reason about the locations of unknown space in the map (Yamauchi 1997; Taylor and Kriegman 1993; Burgard et al. 2000; Yoder and Scherer

The authors gratefully acknowledge the support of ARL Grant W911NF-08-2-0004.

This is one of the several papers published in *Autonomous Robots* comprising the Special Issue on Active Perception.

✉ Erik Nelson
eanelson@berkeley.edu

Micah Corah
micahcorah@cmu.edu

Nathan Michael
nmichael@cmu.edu

[1] Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, 253 Cory Hall, Berkeley, CA 94720, USA

[2] The Robotics Institute, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 12513, USA

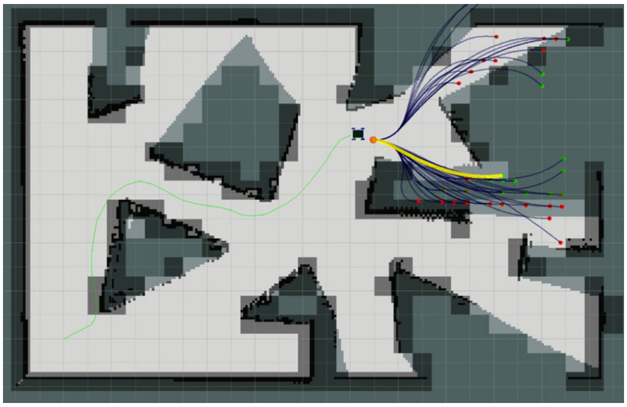


Fig. 1 A robot explores an environment by choosing the action (yellow) that maximizes expected information gain (paths with green endpoints are more informative than paths with red endpoints). Expected information gain is computationally expensive to calculate, so it is instead evaluated with respect to a version of the robot's map that is compressed adaptively according to the structural complexity of the environment (overlaid) (Color figure online)

2015), information-based strategies are more computationally expensive.

In this article, we propose simplifying or compressing a robot's representation of its environment as a means of reducing the computational burden associated with evaluating information-theoretic metrics for active perception planning. The resulting reduction in computational cost allows an exploring robot to evaluate more potential future actions in the same finite amount of time, ultimately allowing it to plan and move more quickly through its environment. Furthermore, we propose an adaptive strategy for determining an optimal map representation to maximize exploration efficiency while minimizing loss of information. The adaptive strategy allows the robot to choose the best map representation based on the structural complexity of its local environment, which could change as the robot navigates. To implement these ideas, we give the robot access to two representations of its own map: a full resolution version that is used for collision checking during planning, and a coarse resolution version that is used to efficiently approximate the informativeness of planned actions (Fig. 1).

The proposed environment model compression strategy leverages techniques from rate distortion theory and signal processing for reducing a random variable to a compressed form while minimizing distortion between the original and reduced representations. Specifically, we use the Information Bottleneck (IB) method (Tishby et al. 2000) to determine a representation for the environment that maximizes compression (therefore maximizing computational efficiency) and minimizes loss of information pertinent to active perception planning (keeping exploration performance relatively unhindered). The IB method is an optimization over possible map representations; in order to generate a set of maps with varying compressions, we apply the Principle of Relevant Information (PRI) (Principe 2010). While both of these methods are general and can be used for many environment models, we demonstrate their application to occupancy grid (OG) maps (Elfes 1989).

Many map compression strategies are proposed in the literature. The OctoMap framework builds an octree data structure to efficiently store the expected occupancy of cells in an environment without allocating memory for a large 3D grid (Wurm et al. 2010). (Im et al. 2010) compress an OG by representing it with wavelets using the Haar wavelet transform. Kretzschmar and Stachniss (2012) compress pose graph maps by examining the mutual information between the pose graph and sensor measurements. Most related to the proposed technique, Einhorn et al. (2011) adaptively choose an OG resolution for individual cells by determining which cells are intersected by measurements. In contrast to these works, we approach environment model compression as one of simultaneously reducing the distortion between the map and its compressed form, and between a compressed map and sensor measurements.

Information-based exploration methods generally choose control actions that seek to maximize the informativeness of future sensor observations, yielding a reduction in map uncertainty computed via metrics such as Shannon mutual information (SMI) (Bourgault et al. 2002; Amigoni and Caglioti 2010; Julian et al. 2013) and Cauchy–Schwarz quadratic mutual information (CSQMI) (Charrow et al. 2015b). We consider information-based strategies instead of geometric strategies (such as frontier exploration) (Yamauchi 1997) because they are generally capable of faster exploration, and do not lose effectiveness when extended to three dimensions (Shen et al. 2011). Similar to the work of Charrow et al. (2015b), we choose to employ CSQMI due to the favorable analytic properties and increased computational efficiency.

Our proposed approach highlights coupling between planning and environment representation and takes advantage of this structure to achieve greatly improved computational efficiency. Investigations at boundary of these problem domains has produced some interesting and related ideas including predicting the structure of unobserved areas for exploration (Strom et al. 2015), and learning the likelihood that an action will cause a collision, given a local map to enable high speed navigation (Richter et al. 2014).

The organization of our presentation follows. We first provide an overview of active perception, several planning strategies for exploration, OG mapping, and relevant ideas from information theory in Sect. 2. The IB method and its application to environment model adaptation is detailed in Sect. 3. The PRI compression method and an example of the PRI applied to OG compression are discussed in Sect. 4. An approach for triggering new environment adaptations in response to the local environment's structure is detailed in Sect. 5. Simulation and experimental evaluation of environment model adaptation during exploration is provided in Sect. 6.

Although the IB and PRI methods for environment model adaptation are described in Nelson and Michael (2015), this article provides a more thorough and comprehensive presentation of the formulations and their implementation throughout. The main additions in this work are a detailed description of active perception planning for mobile robot exploration (Sect. 2), the introduction of OG pyramids for multi-resolution environment model optimization (Sect. 4.3), and considerations for applying the presented methods to exploration in 3D environments (Sect. 6).

2 Active perception planning

In the following section we review methods for planning paths into previously unexplored areas of a robot's map that are likely to result in highly informative sensor measurements, given the robot's current map, sensor model, and dynamics model. These methods will be used in subsequent sections to develop strategies for guiding a robot to locations from which it can optimally reduce uncertainty in its map, and for adapting the robot's map resolution in order to explore more efficiently.

2.1 Active perception as an optimization

In robotics and computer vision, the exercise of choosing control actions that guide a system to locations from which it can gather sensor measurements to learn more about its environment or state is known as active perception. In mobile robotics, one can consider the robot system itself as a sensor that is able to move and actuate for the purpose of collecting informative measurements about the environment. From this perspective, the robot's task is to choose and execute actions that optimize the quality of its sensor measurements. We define an action as a sequence of configurations, $\mathbf{x}_{\boldsymbol{\tau}} \triangleq \{\mathbf{x}_{t+1}, \ldots, \mathbf{x}_{t+T}\}$, that the robot will move to over a future time interval $\boldsymbol{\tau} \triangleq \{t+1, \ldots, t+T\}$. From configurations $\mathbf{x}_{\boldsymbol{\tau}}$, the robot will, in expectation, acquire future sensor measurements $\mathbf{z}_{\boldsymbol{\tau}} \triangleq \{\mathbf{z}_{t+1}, \ldots, \mathbf{z}_{t+T}\}$. Following standard SLAM conventions (Thrun et al. 2005),

$$\mathbf{z}_i = h(\mathbf{x}_i) + \boldsymbol{\eta}_i, \tag{1}$$

where $h(\cdot)$ is the robot's nonlinear measurement equation and $\boldsymbol{\eta}_i \sim \mathcal{N}(\mathbf{0}, \Sigma_i)$ is normally distributed zero-mean measurement noise with covariance matrix Σ_i. In this work we will primarily be concerned with ground robots restricted to configurations that are elements of $\mathbb{SE}(2)$ (the special Euclidean group of 2D transformations), and w.l.o.g. use $\mathbf{x}_i$ to refer to a pose in 2D with first-order dynamics: $\mathbf{x}_i \triangleq (x_i, y_i, \alpha_i, \dot{x}_i, \dot{y}_i, \dot{\alpha}_i)^T$.

In the context of exploring an environment with a mobile robot, the active perception problem can be framed as an optimization over possible future actions that the robot can take,

$$\mathbf{x}_{\boldsymbol{\tau}}^{*} = \underset{\mathbf{x}_{\boldsymbol{\tau}} \in \mathcal{X}_{\boldsymbol{\tau}}}{\operatorname{argmax}}\ \mathcal{J}(\mathbf{m}, \mathbf{z}_{\boldsymbol{\tau}}). \tag{2}$$

Here, $\mathcal{J}$ is a reward function expressing the expected new information learned by sequentially moving the robot to configurations $\mathbf{x}_{\boldsymbol{\tau}}$, collecting sensor measurements $\mathbf{z}_{\boldsymbol{\tau}}$, and updating its map $\mathbf{m}$. $\mathcal{X}_{\boldsymbol{\tau}}$ is the set of all collision-free and dynamically feasible actions that the robot can take over the finite horizon $\boldsymbol{\tau}$, given a deterministic dynamics model $f(\cdot)$,

$$\mathcal{X}_{\boldsymbol{\tau}} = \{\mathbf{x}_{\boldsymbol{\tau}} \mid \mathbf{x}_{i+1} = f(\mathbf{x}_i, \mathbf{u}_i)\ \forall \mathbf{x}_i \in \mathbf{x}_{\boldsymbol{\tau}}\}, \tag{3}$$

where $\mathbf{u}_i$ is a feasible control input. In addition to evaluating the expected pure information content added by $\mathbf{z}_{\boldsymbol{\tau}}$, $\mathcal{J}$ commonly incorporates time or energy expenditure required to carry out the action $\mathbf{x}_{\boldsymbol{\tau}}$.

Unfortunately, the active perception optimization faces the curse of dimensionality; the size of $\mathcal{X}_{\boldsymbol{\tau}}$ grows exponentially with the length of the finite horizon. As $\boldsymbol{\tau}$ increases in size, it therefore quickly becomes infeasible to evaluate $\mathcal{J}$ over all possible actions. Furthermore, the metric itself can have an expensive constant cost per evaluation. To remain computationally tractable, in many cases it is reasonable to generate a subset of candidate actions $\mathscr{X}_{\boldsymbol{\tau}} \subset \mathcal{X}_{\boldsymbol{\tau}}$ that are likely to be informative prior to optimizing (2). If $\mathcal{X}_{\boldsymbol{\tau}}$ is a continuous space, it is likely that the active perception optimization over $\mathscr{X}_{\boldsymbol{\tau}}$ will not choose the most informative action. Broadly speaking, the suboptimality of the solution will depend on the size of $\mathscr{X}_{\boldsymbol{\tau}}$, and therefore will vary inversely with computation time. This is an important consideration, as $\mathscr{X}_{\boldsymbol{\tau}}$ must be expressive enough for exploration, yet must be small for efficiency.

2.2 Occupancy grid mapping

One primary consideration for active perception planning is the model with which the robot should represent its environment. The representation should: (1) allow for fast queries and updates, (2) be expressive enough to allow collision-free trajectory planning, and (3) include a notion of uncertainty. Several environment model options exist, including OGs (Elfes 1989), NDT maps (Saarinen et al. 2013), Gaussian process maps (T O'Callaghan and Ramos 2012; Kim and Kim 2012a), octree variations on OGs (Wurm et al. 2010), and mixture of Gaussian representations (Kim and

Kim 2012b). We opt to employ OGs as they satisfy all three criteria and are simple to work with.

OGs decompose the robot's workspace into a discrete set of 2D or 3D cells with a specified resolution. The presence or absence of obstacles within these cells is modeled as a K-tuple binary random variable $\mathbf{m} \triangleq \{m_i\}_{i=1}^K$, with each m_i taking a value in $\{\texttt{EMP}, \texttt{OCC}\}$, where $\texttt{EMP}$ refers to free space and $\texttt{OCC}$ refers to space occupied by an obstacle. The probability that an individual cell is occupied is given by $o_i \triangleq p(m_i = \texttt{OCC})$, and $1 - o_i = p(m_i = \texttt{EMP})$. The OG representation treats cells as independent from one another, allowing the joint occupancy probability of a specific map to be expressed as the product of individual cell occupancy probabilities: $p(\mathbf{m}) = \prod_{i=1}^K p(m_i)$. Upon initialization, unobserved grid cells are assigned a uniform prior such that $\{o_i = 1 - o_i = 0.5\}_{i=1}^K$. This convention implies that the robot is initially unaware of its surroundings prior to accumulating sensor measurements.

2.3 Cauchy–Schwarz quadratic mutual information

In the exploration task, a robot should execute control actions that minimize uncertainty in its belief distribution over the environment (map). The metric $\mathcal{J}$ must therefore be chosen to capture the effect that sensor measurements have on updates to the map; $\mathcal{J}$ should be larger when $\mathbf{z}_\tau$ contains information that does not already exist in $\mathbf{m}$.

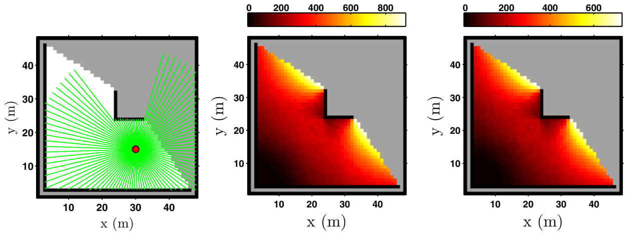
There are many reasonable choices for such a metric. One common choice is SMI (Cover and Thomas 2012; Principe 2010), which is a symmetric binary function used to describe the amount of information one of its arguments contains about the other. The SMI, denoted $I_\mathrm{S}[X; Y]$, between random variables X and Y can be derived from the Kullback–Leibler (KL) divergence,

$$D_{\mathrm{KL}}(q \,||\, r) = \int q(x) \log_2 \frac{q(x)}{r(x)} dx, \quad (4)$$

by setting $q \leftarrow p(X, Y)$ and $r \leftarrow p(X)p(Y)$ (Cover and Thomas 2012). KL divergence is one instance of a more general spectrum of divergences, each of which gives rise to a different mutual information metric (Principe 2010). One divergence measure of particular importance is the Cauchy–Schwarz divergence, which is derived from substituting two distributions, r and q, into the Cauchy–Schwarz inequality,

$$D_{\mathrm{CS}}(q \,||\, r) = \log_2 \frac{\int q^2(x)dx \int r^2(x)dx}{\left(\int q(x)r(x)dx\right)^2}. \quad (5)$$

Similar to the derivation of SMI from KL divergence, setting $q \leftarrow p(X, Y)$ and $r \leftarrow p(X)p(Y)$ as the arguments of CS divergence gives rise to the Cauchy–Schwarz Quadratic Mutual Information (CSQMI). Like SMI, CSQMI is non-negative, symmetric, and zero only when its arguments are independent (i.e. $p(X, Y) = p(X)p(Y)$). Importantly for exploration, CSQMI can be computed analytically, and is more efficient to compute than SMI when using a beam-based sensor model on an OG (Charrow et al. 2015b). The fact that CSQMI can be computed analytically is critical; SMI often cannot be computed analytically and requires expensive numerical approximations such as Monte Carlo sampling or numerical integration proposed by Julian et al. (2013) in an analagous formulation. As a result, CSQMI can be up to one order of magnitude faster to compute (Charrow et al. 2015b). Figure 2 shows that the CSQMI and SMI between a beam-based measurement and an OG produce similar values. Given the benefits of CSQMI, it is a suitable choice for the metric $\mathcal{J}$ in the active perception optimization (2). The CSQMI between an OG map and a sequence of beam-based measurements, $I_{\mathrm{CS}}[\mathbf{m}; \mathbf{z}_\tau \mid \mathbf{x}_\tau]$, is



(a) **(b)** **(c)**

Fig. 2 Two variants of mutual information (**b**: SMI; **c**: CSQMI) densely computed in free space over an occupancy grid **a** using a 100-beam omnidirectional 2D LiDAR with 30 m range. An example sensor measurement is depicted in **a**. Controlling the robot towards locations that maximize either variant of mutual information would attract the robot to locations from which it could observe unknown areas of the map

$$\log_2 \frac{\int \sum_{\mathcal{M}} p^2(\mathbf{m}, \mathbf{z}_\tau) d\mathbf{z}_\tau \int \sum_{\mathcal{M}} p^2(\mathbf{m}) p^2(\mathbf{z}_\tau) d\mathbf{z}_\tau}{\left(\int \sum_{\mathcal{M}} p(\mathbf{m}) p(\mathbf{z}_\tau) p(\mathbf{m}, \mathbf{z}_\tau) d\mathbf{z}_\tau\right)^2}, \quad (6)$$

where $\mathcal{M}$ is the set of all K-cell maps. In general, a map can take one of $|\mathcal{M}| = 2^K$ values. However, only areas of intersection between beams in measurements $\mathbf{z}_\tau$ and grid cells in the map yield non-zero values. We follow the method in Charrow (2015) for computing CSQMI (which includes a derivation starting from Eq. (6) that is too lengthy to include here, but results in an easy-to-implement algorithmic form). This method computes CSQMI to a close approximation in $\mathcal{O}(n)$ time, or exactly in $\mathcal{O}(n^2)$ time, where n is the number of grid cells intersected by measurements in $\mathbf{z}_\tau$.

Substituting $I_{\mathrm{CS}}[\mathbf{m}; \mathbf{z}_\tau \mid \mathbf{x}_\tau]$ for $\mathcal{J}(\mathbf{m}, \mathbf{z}_\tau)$ in (2) yields an active perception optimization that guides a robot toward unexplored locations by choosing control actions that maximally reduce uncertainty in the map.

2.4 Receding horizon planning and action generation

Algorithm 1 ActivePerceptionPlanning($\mathcal{J}(\cdot,\cdot)$, $\mathbf{m}$, $\boldsymbol{\tau}$)

```
1: while MapIncomplete($\mathbf{m}$) do
2:     $\mathcal{X}_{\boldsymbol{\tau}} \leftarrow$ GenerateActions($\mathbf{m}$, $\boldsymbol{\tau}$)
3:     $\mathbf{x}^*_{\boldsymbol{\tau}} \leftarrow$ ActivePerceptionOptimization($\mathcal{J}(\cdot,\cdot)$, $\mathcal{X}_{\boldsymbol{\tau}}$, $\mathbf{m}$)
4:     repeat
5:         Follow($\mathbf{x}^*_{\boldsymbol{\tau}}$)
6:         $\mathbf{z} \leftarrow$ Sense()
7:         $\mathbf{x} \leftarrow$ Localize($\mathbf{m}$, $\mathbf{z}$)
8:         $\mathbf{m} \leftarrow$ UpdateMap($\mathbf{m}$, $\mathbf{x}$, $\mathbf{z}$)
9:     until ReplanTimerExpired()
10: end while
```

The goal of action generation is to build a set $\mathcal{X}_{\boldsymbol{\tau}}$ of dynamically feasible actions, from which the robot can execute the most informative. In this article we consider the use of receding horizon control strategies, where action generation is repeated during each planning iteration, and the robot only carries out a portion of each action before replanning. This control strategy is used because the robot's map is continuously updated with new sensor measurements, sometimes rendering the current plan infeasible due to newly discovered collisions with obstacles in the environment. The interaction between algorithmic components is detailed in Algorithm 1.

Several options exist for generating actions. An optimization based method for efficiently identifying continuous informative trajectories is proposed by Marchant and Ramos (2014), which uses a Gaussian Process map representation. Other recent works by Charrow et al. (2015b), Charrow et al. (2015a) and Vallvé and Andrade-Cetto (2014) suggest seeding action generation by identifying frontiers (Yamauchi 1997) in the map and evaluating $\mathcal{J}$ near frontier locations. Once the most informative frontier is identified, actions can be planned towards that frontier using, e.g. an RRT (LaValle 1998) or one of its many variants. Because frontier identification is efficient, this two-pass approach is useful for finding potentially informative locations prior to performing the comparatively expensive reward evaluation step. This strategy has the added benefit that frontiers are computed globally across the robot's map, guaranteeing that the robot won't become trapped in a dead-end or a pose from which its local map is already fully explored. After planning global actions towards frontiers, an optional optimization step can be applied to increase expected information gain along the path (Charrow et al. 2015a; Kollar and Roy 2008). More thorough surveys of frontier exploration algorithms and heuristics are provided by Basilico et al. (2008) and Holz et al. (2011). One unfortunate downside of frontier exploration is that the approach does not extend well to 3D, where frontiers are generally smaller and far more numerous (Shen et al. 2011).

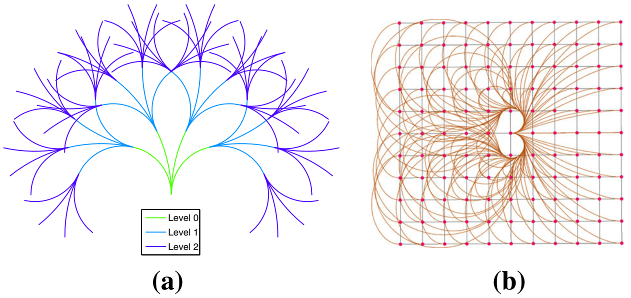


Fig. 3 Two strategies for action generation: a primitive library with three levels, built from a dictionary of four motion primitives (**a**), and an 11 × 11 × 1 lattice graph generated by solving many BVPs from the robot's initial state (middle, facing right) to a lattice of final states (**b**). **a** Motion primitive library, **b** lattice graph

Actions can also be generated by sampling from a set of motion primitives (Fig. 3). Since motion primitives can be precomputed prior to online exploration, they are an efficient choice for real-time exploration. Collision checking involves stepping along actions during a breadth-first or depth-first search and pruning all nodes (actions) that lie below those that contain a collision.

Finally, a third method for generating actions is lattice graph planning. Lattice graph planners define a discrete set of goal states, and solve Boundary Value Problems (BVPs) to find trajectories from the origin to each goal (Pivtoraiko and Kelly 2005; Pivtoraiko et al. 2009, 2013) (Fig. 3). The resulting set of motion primitives can be rotated and translated to the robot's current position at run-time, and sampled from to produce candidate actions. Like other motion primitives, lattice graph actions can be precomputed and are therefore a suitable choice for real-time exploration.

The latter two action generation strategies produce local plans, and do not consider globally informative goal locations. We do however utilize these myopic planners throughout our discussion to demonstrate the computational savings of the proposed map compression strategies, noting that a global planning strategy can always be added on top.

3 Environment model adaptation using the information bottleneck method

To rapidly explore an uncertain environment, an autonomous mobile robot must update its localization, mapping, planning, and control loops at a high frequency. When navigating at a high velocity, the planning step (specifically, line 3 of Algorithm 1) is comparatively expensive; an increased velocity requires increased planning frequency, yet one iteration of planning incurs the fixed computational cost of evaluating the active perception cost function in (2) over the set $\mathcal{X}_{\boldsymbol{\tau}}$ of candidate actions. Regardless of the information metric chosen, for a fixed $|\mathcal{X}_{\boldsymbol{\tau}}|$ there is a threshold planning frequency

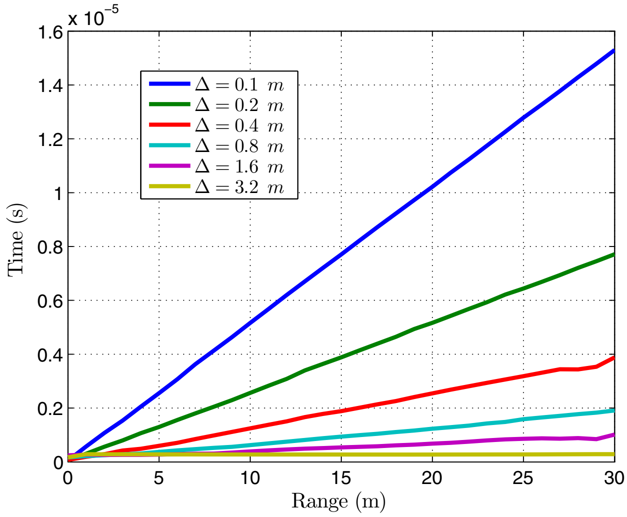


Fig. 4 Time (median of 10^5 samples, Intel 2.7 GHz processor) to evaluate CSQMI for a single beam is empirically linear in both the OG cell resolution Δ, and the measurement range.

past which the robot will saturate its computational resources. This threshold restricts the velocity of the robot's navigation, and therefore the time efficiency of its exploration.

For example, Fig. 4 demonstrates that the time required to evaluate CSQMI (i.e. evaluate line 3 of Algorithm 1) for one beam on an OG map at 30 m is 0.015 ms at 10 cm cell resolution. Computing CSQMI over a subsampled 2D laser scan with 500 beams would therefore require 7.5 ms, limiting a 10 Hz single-threaded planner to consideration of only 13 actions at a time, which is insufficient for high speed indoor navigation.

Fortunately, the computational cost of active perception planning can be dramatically reduced by approximating $\mathcal{J}$. In this section we propose a method for reducing the cost of (2) by taking advantage of the way CSQMI (or SMI, if desired) is calculated for beam-based sensors. The approximation involves simplifying or compressing the environment model, $\mathbf{m}$, in a way that sacrifices a minimal amount of information pertinent to the cost function $I_{\text{CS}}[\mathbf{m}; \mathbf{z}_\tau \mid \mathbf{x}_\tau]$.

Calculating information metrics between an uncertain map and an expected future sensor measurement (e.g. SMI and CSQMI) requires effectively simulating beams that would be captured from locations along $\mathbf{x}_\tau$ and evaluating the amount of new information that those measurements would be expected to uncover given the current map. With an OG representation, evaluating these metrics therefore requires iterating over beams in the simulated measurement, raycasting along those beams to find intersections with cells in the map, and computing an expected reduction in uncertainty using the intersected cells' occupancy values. Intuitively, as the resolution of cells in the OG decreases, so does the number of cells that a raycast must traverse. Therefore the computational cost of information-theoretic reward evaluation is a function of the cell resolution of the OG. For example, by employing the approximate CSQMI technique proposed by Charrow et al. (2015b), computational cost becomes linear in both the OG cell resolution and the range of the measurement (Fig. 4). Reducing the number of cells needed to express the OG while preserving the occupancy information it contains would therefore lead to large computational savings.

3.1 The information bottleneck method

Lossily compressing the environment model will necessarily lead to distortion in the values returned by CSQMI, the excess of which will cause the active perception optimization (2) to choose a poor action for exploration. It is therefore prudent to balance compression of the environment model (fast CSQMI evaluation) with conservation of information (good action choices). To this end, we employ the Information Bottleneck (IB) method from rate distortion theory, a method for finding the optimal compressed representation $\hat{X}$ of a random variable X that preserves maximum information about a second random variable Y,

$$\min_{p(\hat{x} \mid x)} \; I[X; \hat{X}] - \beta I[\hat{X}; Y]. \tag{7}$$

Here, $I[\cdot\, ; \,\cdot]$ is a general information metric operating on the distributions of two random variables (for instance, SMI or CSQMI). The IB optimization provides a trade-off between compressing the variable X and preserving meaningful information through the tuning parameter $\beta \in \mathbb{R}_+$. As $\beta \to 0$, the optimization tends towards the trivial point compression, whereas when $\beta \to \infty$, $\hat{X}$ approaches X (Principe 2010). The two variables in the information terms of the IB functional can be equivalently thought of as squeezing the information that X contains about Y through a "bottleneck" quantization function $Q(\cdot)$ (where $\hat{X} = Q(X)$) (Tishby et al. 2000) (Fig. 5).

The IB method permits an explicit solution for the $p(\hat{x} \mid x)$ minimizing (7) when using SMI as the information metric. However, the solution is computed iteratively and therefore requires multiple evaluations of the distributions $p(\hat{x} \mid x)$, $p(\hat{x})$, and $p(y \mid \hat{x})$ (Tishby et al. 2000).

3.2 Map adaptation via the IB method

To tailor the IB method to the problem of robot exploration, let X refer to the robot's map $\mathbf{m}$, and Y to measurements $\mathbf{z}_\tau$. The IB method would ideally enable adaptive selection of a simplified map representation as the robot navigates. In this case, the iterative algorithm for computing the optimal solution is too expensive for online scenarios. Instead, we find the map compression function, C_*, from a family of

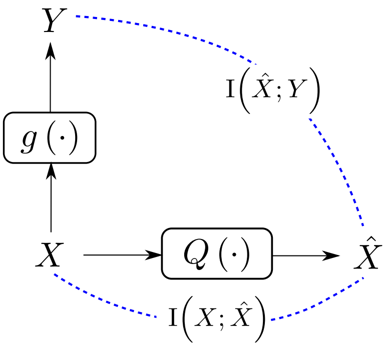
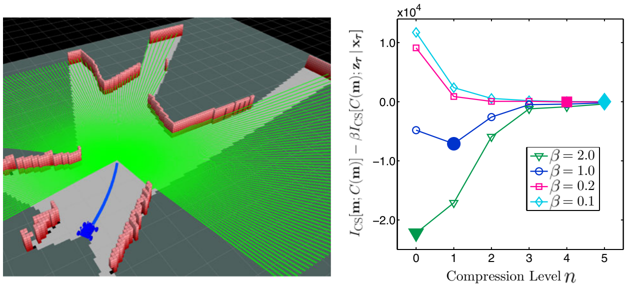


Fig. 5 A diagram of variables relevant to the IB method. IB attempts to minimize the mutual information between X and its compressed form, $\hat{X}$, while maximizing the mutual information between X and a second variable, $Y = g(X)$. Q is a function quantizing X



(a) **(b)**

Fig. 6 **a** A sensor measurement is simulated from the endpoint of a planned action. The IB cost functional in (8) is shown on the dependent axis of **b** for varying values of β. The optimal compression level (filled markers) decreases as β increases, favoring preservation of information about the measurement

compressions, $\mathcal{C}$, that minimizes the IB functional

$$C_* = \underset{C \in \mathcal{C}(\mathbf{m})}{\operatorname{argmin}}\ I_{\mathrm{CS}}[\mathbf{m}; C(\mathbf{m})] - \beta I_{\mathrm{CS}}[C(\mathbf{m}); \mathbf{z}_\tau \mid \mathbf{x}_\tau]. \quad (8)$$

Note that the IB method does not require the environment model $\mathbf{m}$ to be an OG, nor does it require any specific compression operation on $\mathbf{m}$. In this sense, the IB method generalizes across choice of environment model and choice of compression on that environment model.

The influence of the parameter β on the IB optimization is shown for a multi-beam measurement captured from a planned future location in Fig. 6. The IB cost functional is plotted for varying values of β. When β is small, the optimization is dominated by the minimum information term and favors maximum compression, whereas when β is large, the optimization favors preservation of information.

During exploration, adapting the robot's environment model requires evaluation of $I_{\mathrm{CS}}[\mathbf{m}; C(\mathbf{m})]$ and $I_{\mathrm{CS}}[C(\mathbf{m}); \mathbf{z}_\tau \mid \mathbf{x}_\tau]$ over $|\mathcal{C}|$ versions of the map. In Sect. 4 we describe a strategy for generating the set $\mathcal{C}$ of compression functions, and in Sect. 5 we propose an approach for triggering the IB optimization adaptively to decide on the best current environment representation.

4 Environment model compression using the principal of relevant information

The IB method from Sect. 3 can be used to select an environment representation that retains a maximum amount of information useful to the active perception optimization (2), and simultaneously minimizes its computational cost. The IB method assumes access to a family of functions, $\mathcal{C}$, that operate on the map to produce a compressed version, or in the case of OGs, a version with a lower resolution.

In this section we review the Principal of Relevant Information (PRI) (Principe 2010), a second technique from rate distortion theory, for compressing a robot's environment model. We then apply the IB method and the PRI to OG maps, and introduce the concept of an OG pyramid. We show that in many cases the resolution of an OG can be dramatically reduced before exploration performance is negatively impacted.

4.1 The principle of relevant information

The problem of reducing an environment model to its relevant information (Geiger and Kubin 2013) can be formulated as an information theoretic optimization using the PRI. The PRI is a technique for learning a reduced representation $\hat{X}$ of a random variable X such that both the entropy of $\hat{X}$ and the divergence of $\hat{X}$ with respect to X are minimized.

$$\Lambda(\hat{X}) = \min_{\hat{X}}\ H(\hat{X}) + \lambda D(X \,||\, \hat{X}). \quad (9)$$

Here, $H(\cdot)$ is a general entropy measure, and $D(\cdot \,||\, \cdot)$ is a general divergence measure. The PRI cost function shares many similarities with the IB optimization in (7). The two terms of the PRI cost function are Rényi's α-entropy, a generalization of Shannon's entropy that describes the amount of uncertainty in its argument, and Rényi's α-divergence, another generalized divergence measure that describes the distortion between $p(x)$ and $p(\hat{x})$. These terms simplify to Shannon entropy and KL divergence for $\alpha = 1$.

Intuitively, the PRI trades off information redundancy in $\hat{X}$ (minimize entropy) for errors induced by using the compressed form $\hat{X}$ to represent the original (minimize divergence). The variational parameter $\lambda \in \mathbb{R}_+$ balances this trade-off. Choosing $\lambda = 0$ forces the optimization to select $\hat{X}$ such that $H(\hat{X}) = 0$. Total entropy minimization is only possible for values of $\hat{X}$ that are completely determined. By contrast, choosing $\lambda \to \infty$ reduces to minimizing the divergence between $p(x)$ and $p(\hat{x})$, giving back the original data.

Following Principe et al. (2010), we choose $\alpha = 2$ which gives us Rényi's 2-entropy,

$$H_2(X) = -\log_2 \sum_i p^2(x_i), \tag{10}$$

and the Cauchy–Schwarz divergence (5), which together allow a direct relation between the two terms of the PRI cost function,

$$\begin{aligned} &H_2(\hat{X}) + \lambda D_{\mathrm{CS}}(X \,||\, \hat{X}) \\ &\quad = (1-\lambda) H_2(\hat{X}) - \lambda H_2(X) - 2\lambda \log_2 \sum_i p(\hat{x}_i) p(x_i). \end{aligned} \tag{11}$$

The second term has no influence on the minimization over $\hat{X}$, and can be ignored. To simplify the optimization, we choose to give equal weight to entropy and divergence, and optimize for $\lambda = 1$. By noting that the summand in the third term of (11) must be positive, the PRI optimization can be simplified to

$$\Lambda(\hat{X}) = \max_{\hat{X}} \sum_i p(\hat{x}_i) p(x_i). \tag{12}$$

4.2 Using the PRI for occupancy grid compression

The PRI is well-suited for simplifying a robot's environment model. We now provide details of customizing the PRI for OG compression as an example.

Ideally, a low-resolution compressed OG would represent its high-resolution originator well. The divergence minimization term in the PRI accomplishes this. The entropy term in the PRI drives occupancy values in the compressed map towards being determined. For example, briefly suppose that a 2×1 region of cells must be compressed to a single cell, where one of the original cells has a high probability of occupancy and the other has a low probability. Minimizing Cauchy–Schwarz divergence would result in a mixture of the two probability values, which increases the uncertainty of the cell's occupancy. While this increase in uncertainty through compression is instinctive in most applications, robot navigation typically relies on operations such as raycasting and collision checking on the map. For these operations it would be harmful to lose information about free and occupied regions of the environment that were known prior to compression. Minimizing entropy in the PRI optimization alleviates these concerns, as it forces cell occupancy probabilities in the compressed map towards being determined. In other words, the entropy minimization component of the PRI cost function forces occupancy probabilities towards EMP and OCC.

To apply the PRI to OG compression, let $\mathbf{m}^K$ be an OG with K cells (for the remaining map notations, superscripts will denote cell counts), and substitute it for X. In order to solve the problem, we add three constraints.

1. $\hat{X}$ should represent X well locally. Compression over the entire map can therefore be accomplished by performing compression in many small square (cubic in 3D) independent regions $\mathbf{m}^R \subseteq \mathbf{m}^K$ by exercising the OG assumption that individual cell occupancy probabilities are independent.
2. To remain computationally tractable, only the set of compressions that reduce OG cell count in each dimension by factors of two will be considered. Therefore an OG $\mathbf{m}^K$ will be compressed to an OG $\mathbf{m}^{2^{-dn}K}$, where d is the OG dimension and n is the number of 2× compressions in each dimension. The set of compressions with this property can be expressed using

   $$C_n(\mathbf{m}^K) = \mathbf{m}^{2^{-dn}K}, \quad n \in \mathbb{N}_{\geq 0}, \tag{13}$$

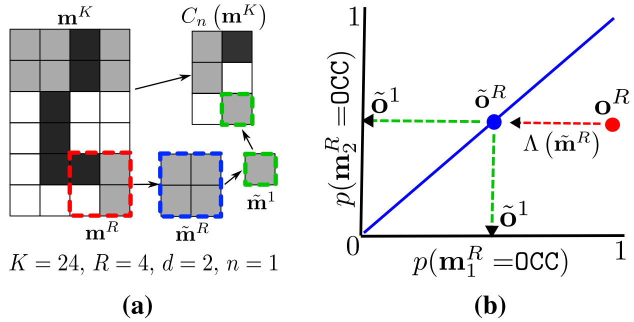
   remembering that when $n = 0$, $C_n(\mathbf{m}^K) = \mathbf{m}^K$. The value of $2^{-dn}K$ may not be integer if the cell count along any dimension in the original map is not a power of 2. In practice, edge cases can be handled with padding, but for this discussion we consider only maps with a cell count that is a power of two in each dimension. Both $\mathbf{m}^K$ and $C_n(\mathbf{m}^K)$ will have the same global metric dimensions, but will have different cell edge lengths and cell counts when $n \geq 1$.
3. If X is an OG, $\hat{X}$ must also be an OG.

Under these constraints, the map can be compressed by decomposing it into independent regions, and compressing each region. For each region $\mathbf{m}^R$, the PRI can be used to find a multivariate random variable $\tilde{\mathbf{m}}^R$ that has uniform occupancy probabilities and minimizes both entropy and divergence with respect to $\mathbf{m}^R$. The occupancy probability of each $\tilde{\mathbf{m}}^R$, which we denote $\tilde{o}^R \triangleq p(\tilde{\mathbf{m}}^R = \{\texttt{OCC}, \ldots, \texttt{OCC}\}) = p(\tilde{\mathbf{m}}_i^R = \texttt{OCC}), \forall i \in \{1, \ldots, R\}$, can be reduced to a scalar, yielding the occupancy probability of a single cell in the compressed OG: $\tilde{o}^1 \triangleq p(\tilde{\mathbf{m}}^1 = \texttt{OCC})$ (Fig. 7). The occupancy distribution of a cell in the compressed map is completely determined by knowing $\tilde{o}^1$, because $p(\tilde{\mathbf{m}}^1 = \texttt{OCC}) = 1 - p(\tilde{\mathbf{m}}^1 = \texttt{EMP})$, so the set of $\tilde{o}^1$ values from independent regions are all that is necessary to determine the compressed OG $C_n(\mathbf{m}^K)$.

Using these notations alongside the PRI optimization in (9), an OG region $\tilde{\mathbf{m}}^R$ that minimizes entropy and divergence with respect to $\mathbf{m}^R$ can be found by maximizing

$$\Lambda(\tilde{\mathbf{m}}^R) = \max_{\tilde{\mathbf{m}}^R} \sum_{\mu^R} p(\tilde{\mathbf{m}}^R = \mu^R) p(\mathbf{m}^R = \mu^R), \tag{14}$$

where μ^R iterates over maps of size R.



$\mathbf{m}^K$
$C_n\left(\mathbf{m}^K\right)$
$\tilde{\mathbf{m}}^1$
$\mathbf{m}^R$
$\tilde{\mathbf{m}}^R$
$K = 24,\ R = 4,\ d = 2,\ n = 1$
(a)
1
$p(\mathbf{m}_2^R = \texttt{OCC})$
$\tilde{\mathbf{o}}^1$
$\tilde{\mathbf{o}}^R$
$\mathbf{o}^R$
$\Lambda\left(\tilde{\mathbf{m}}^R\right)$
$\tilde{\mathbf{o}}^1$
0
$p(\mathbf{m}_1^R = \texttt{OCC})$
1
(b)

Fig. 7 For each square (cubic in 3D) region $\mathbf{m}^R$ in the uncompressed OG $\mathbf{m}^K$, the PRI optimization finds a random variable $\tilde{\mathbf{m}}^R$ that minimizes (9) and is constrained to have uniform occupancy probability. **a** OG compression sequence, **b** Probability space of the top two grid cells in $\mathbf{m}^R$ in **a**

Solving this optimization involves iterating through all permutations of maps that have cell count R and multiplying the probability that $\mathbf{m}^R$ takes on a specific permutation, μ^R, with the probability that $\tilde{\mathbf{m}}^R$ also takes on that permutation. Fortunately, although the space of such maps is large, $p(\tilde{\mathbf{m}}^R)$ is zero for all but two permutations due to the constraint that $\tilde{\mathbf{m}}^R$ must have uniform occupancy. These two permutations are the maps $\{\texttt{EMP}, \ldots, \texttt{EMP}\}$ and $\{\texttt{OCC}, \ldots, \texttt{OCC}\}$. In all other permutations, the variable $p(\tilde{\mathbf{m}}^R = \mu^R)$ evaluates to zero, causing the corresponding term in the sum to evaluate to zero. The two non-zero terms in the summand of (16) can therefore be enumerated explicitly,

$$\begin{aligned}
&\sum_{\mu^R} p(\tilde{\mathbf{m}}^R = \mu^R) p(\mathbf{m}^R = \mu^R) \\
&\quad = p(\tilde{\mathbf{m}}^R = \{\texttt{EMP}, \ldots, \texttt{EMP}\}) p(\mathbf{m}^R = \{\texttt{EMP}, \ldots, \texttt{EMP}\}) \\
&\qquad + p(\tilde{\mathbf{m}}^R = \{\texttt{OCC}, \ldots, \texttt{OCC}\}) p(\mathbf{m}^R = \{\texttt{OCC}, \ldots, \texttt{OCC}\}) \\
&\quad = (1 - \tilde{o}^1) \prod_{i=1}^{R} (1 - o_i^R) + \tilde{o}^1 \prod_{i=1}^{R} o_i^R. \qquad (15)
\end{aligned}$$

Rather than finding the value of $\tilde{\mathbf{m}}^R$ that maximizes the cost function (i.e. the compressed map region itself), we are more interested in the occupancy probability of that map region. We therefore modify the optimization in (14) to return the argument of maximization

$$\tilde{\mathbf{m}}_*^R = \underset{\tilde{\mathbf{m}}^R}{\operatorname{argmax}} \sum_{\mu^R} p(\tilde{\mathbf{m}}^R = \mu^R) p(\mathbf{m}^R = \mu^R), \qquad (16)$$

noting that the occupancy value of the compressed map region can be recovered with $\tilde{o}_*^1 = \tilde{o}_*^R = p(\tilde{\mathbf{m}}_i^R = \texttt{OCC}), \forall i \in \{1, \ldots, R\}$.

Substituting the expanded sum into the PRI optimization in (16) gives

$$\tilde{\mathbf{m}}_*^R = \underset{\tilde{\mathbf{m}}^R}{\operatorname{argmax}} \left((1 - \tilde{o}^1) \prod_{i=1}^{R} (1 - o_i^R) + \tilde{o}^1 \prod_{i=1}^{R} o_i^R \right). \qquad (17)$$

Enumerating the two possibilities: when all of the cells in $\tilde{\mathbf{m}}^R$ are empty, the objective function evaluates to $\prod_{i=1}^R (1 - o_i^R)$, and when all of the cells are occupied, the objective evaluates to $\prod_{i=1}^R o_i^R$. Denoting $\pi^R \triangleq \prod_{i=1}^R \frac{o_i^R}{1 - o_i^R}$, these two possibilities imply that the solution to Eq. (17) is

$$\tilde{\mathbf{m}}_*^R = \begin{cases} \{\texttt{EMP}, \ldots, \texttt{EMP}\} & \text{if } \pi^R < 1 \\ \{\texttt{OCC}, \ldots, \texttt{OCC}\} & \text{if } \pi^R > 1 \end{cases}. \qquad (18)$$

with an ambiguous case when $\pi^R = 1$, which is the case when all cells in the uncompressed map region either have exactly contradictory values, or have an unknown occupancy. Knowing the occupancy status of the compressed map region allows us to assign an occupancy probability to each case.

$$\tilde{o}_*^1 = \begin{cases} 0 & \text{if } \pi^R < 1 \\ 1 & \text{if } \pi^R > 1 \\ \frac{1}{2} & \text{if } \pi^R = 1 \end{cases}. \qquad (19)$$

The PRI solution for OG compression in (19) yields a simple rule: if the product of cell occupancy likelihoods in a given region is greater than one, set the occupancy probability of the corresponding cell in the compressed OG to one. Likewise set the occupancy probability of the single cell to zero if π^R is less than one, and to one half if π^R evaluates to one. Note that the third case occurs when either all cells in the uncompressed map have occupancy probabilities that average to one half. This includes the case when their occupancies are all exactly equal to one half. Therefore fully unknown regions and regions of mixed certainty compress to unknown cells.

While the optimal PRI solution gives reasonable compressed maps, one may introduce a heuristic to increase the fraction of occupied cells that are preserved through compression by multiplying the right-hand sides of the inequalities in (19) by $\eta \in \mathbb{R}_+$. As η decreases, occupied cells will be preserved through compression with higher frequency. For example, for applications involving raycasting it can be useful to include this heuristic so that occupied cells in the original map do not vanish through compression and lead to longer raycasts. This heuristic modification can be applied by instead using

$$\tilde{o}_*^1 = \begin{cases} 0 & \text{if } \pi^R < \eta,\ \pi^R \neq 1 \\ 1 & \text{if } \pi^R \geq \eta,\ \pi^R \neq 1 \\ \frac{1}{2} & \text{if } \pi^R = 1. \end{cases} \qquad (20)$$

Note that when using the η heuristic, the $\pi^R = 1$ condition is kept so that completely unknown square regions of cells

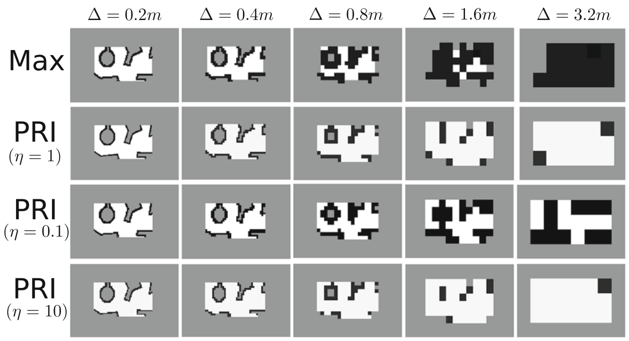


Fig. 8 A small map compressed using four different strategies. As cell edge length Δ grows, the PRI strategy retains map geometry for longer than maximum occupancy compression

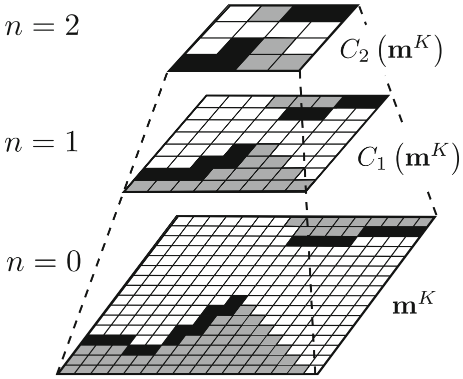


Fig. 9 A three-level OG pyramid.

remain unknown through compression. We advise using $\eta < 1$ for applications involving raycasting (e.g. exploration), and $\eta = 1$ for general purpose map compression.

The PRI compression solution is compared in Fig. 8 for different values of η against maximum occupancy compression (used by other environment models such as OctoMap (Wurm et al. 2010). As the amount of compression increases, the PRI solution tends to retain major map features. Decreasing the heuristic parameter η from one causes occupied cells to be retained through compression with a higher frequency.

4.3 Occupancy grid pyramids

The IB method from Sect. 3 assumes access to a family of functions, $\mathcal{C}$, that operate on a representation of the environment to produce a compressed version. The IB method selects one function with which to compress the map, yielding a low resolution representation that can then be used to more efficiently evaluate CSQMI and other information metrics for exploration.

Armed with the PRI optimization, we define a multi-resolution OG pyramid on the map $\mathbf{m}$ as

$$\mathcal{C}_n(\mathbf{m}) \triangleq \{C_i(\mathbf{m})\}_{i=0}^{n}, \quad n \in \mathbb{N}_{\geq 0}, \tag{21}$$

where each OG $C_i(\mathbf{m})$ is generated by applying the PRI optimization in (20) to $\mathbf{m}$ over larger and larger square or cubic regions. The data structure's name is in reference to the image pyramid, a multi-resolution image representation commonly used in computer vision (Burt and Adelson 1983). A three-level OG pyramid is depicted in Fig. 9.

5 Adapting the environment model online

To aggregate the ideas introduced in Sects. 2–4, we now outline a strategy for adapting the robot's environment model during exploration in response to its local environment. The goal of online environment model adaptation is to reduce the robot's map down to a minimum amount of information necessary for exploration, so that the active perception optimization in (2) can be computed rapidly. These factors enable higher planning frequencies and consideration of more actions per planning iteration, enabling increased rates of exploration. For the following formulations we assume an OG environment model. However the environment does not necessarily need to be represented by an OG, as the IB method and the PRI compression strategy generalize to other environment representations.

During each planning iteration, the robot is given access to two versions of its map: one full resolution version that is updated with new sensor measurements with line 8 of Algorithm 1, and a coarser resolution map that is computed by solving the IB optimization in (8). The full resolution map is used for collision checking along planned actions which in our case is a relatively inexpensive process that must be performed at the maximum available resolution. The coarse resolution map is used for evaluating CSQMI efficiently.

To reinforce the notion of computational savings afforded by map compression, Fig. 4 demonstrates that using $C_n(\mathbf{m}^K)$ in place of $\mathbf{m}^K$ will reduce the time required to evaluate CSQMI by 2^n (CSQMI is computed by summing over cells intersected by 1D beams). In Sect. 6 we will show that OG maps can typically be compressed to cells on the order of 0.8–3.2 m before exploration performance is significantly altered. Substituting these for an OG with a base resolution of 0.1 m, a typical choice for indoor mapping, would result in a worst case CSQMI cost reduction of 8 times, and a best case reduction of 32 times.

The IB optimization (8) is used to select a resolution for the coarse map. Rather than solving the optimization after the first several sensor measurements are integrated into the map and fixing the resulting environment representation, we employ an adaptive strategy that recomputes an optimal compression function when the robot enters a significantly different area of its environment. Adaptation is useful when, for example, the robot transitions from a wide-open area consisting of mostly empty cells that can be heavily compressed without discarding much information to a narrow corridor that cannot.

Algorithm 2 AdaptEnvironmentModel($\mathbf{m}, \boldsymbol{\tau}, n, \delta$)

```
1: $h \leftarrow$ MeanLocalMapEntropy($\mathbf{m}$)                          ▷ Eq. (10)
2: if $|h - h_{\text{last}}| \geq \delta$ then
3:     $\mathcal{C}_n(\mathbf{m}) \leftarrow$ BuildOGPyramid($\mathbf{m}, n$)        ▷ Eq. (21)
4:     $\mathscr{X}_{\boldsymbol{\tau}} \leftarrow$ GenerateActions($\mathbf{m}, \boldsymbol{\tau}$)
5:     $C_*(\mathbf{m}) \leftarrow$ EvaluateIB($\mathcal{C}_n(\mathbf{m}), \mathscr{X}_{\boldsymbol{\tau}}$)        ▷ Eq. (8)
6:     $h_{\text{last}} \leftarrow h$
7: end if
```

To adapt the environment model, on a fixed frequency the robot evaluates the mean entropy of cells in a fixed size local submap around its current position using (10), giving a value in the range [0, 1]. Although not a perfect indicator, a large increase or decrease in local map entropy signifies that the local environment has changed in structure. Other possible triggers for adaptation include the mean or minimum distance to an obstacle, changes to the ratio of free space to occupied space in the local map, or changes to feature-based map descriptors. If the absolute change in entropy since the last adaptation is greater than a threshold amount, an OG pyramid is generated using the PRI compression method, and an IB optimization is triggered on the new OG pyramid to select an optimal representation for the map. The environment model adaptation process is outlined in Algorithm 2.

The new coarse resolution map is updated and used alongside the full resolution map for CSQMI evaluation until the criteria in line 2 of Algorithm 2 is met again. Calculating the entropy of cells in a local submap around the robot has a bounded computational cost, and is inexpensive in comparison to the active perception optimization (2) that is performed each planning iteration. While the IB optimization itself is expensive (as CSQMI must be computed over n different maps of varying resolution), the condition for entering the adaptation subroutine occurs infrequently.

6 Results

6.1 Effects of map compression on exploration

Environment model compression was motivated originally by the observation in Fig. 4 that compression results in increased efficiency of evaluating CSQMI, in turn reducing planning time, and allowing a robot to explore its environment at a higher velocity. However, it is important to ensure that sensor measurements deemed informative on the original map remain informative after compression. A library of forward-arc motion primitives (Sect. 2.4) planned from a simulated robot's position into a partially explored map is depicted in Fig. 10. Reward is computed at the actions' endpoints using CSQMI between the expected future sensor measurement and compressed maps. The relative rewards offered by the planned actions retain their ordering until the map is compressed significantly ($n = 5$), at which point a different optimal action is chosen. Although different, the optimal action chosen for the map with the highest compression is still a high-reward path with respect to the original map. Most importantly, computing CSQMI reward on the map with the highest compression is 32 times more efficient than on the original map. To demonstrate the PRI compression strategy, 2D and 3D OG pyramids of a large warehouse environment are shown in Figs. 11 and 12. The parameter η was set to 0.1 for the 2D pyramid, and to 0.5 for the 3D pyramid to persuade the compression to retain occupied cells. This preservation is clearly demonstrated in the transition from Fig. 12a–d, where single occupied cells surrounded by free areas in the original map remain occupied even after reducing resolution from 0.1 m to 0.8. In the 2D map, intersections between free and unknown space are preserved until a severe compression is applied ($n = 5$). Regions occupied by obstacles are represented well throughout all map resolutions.

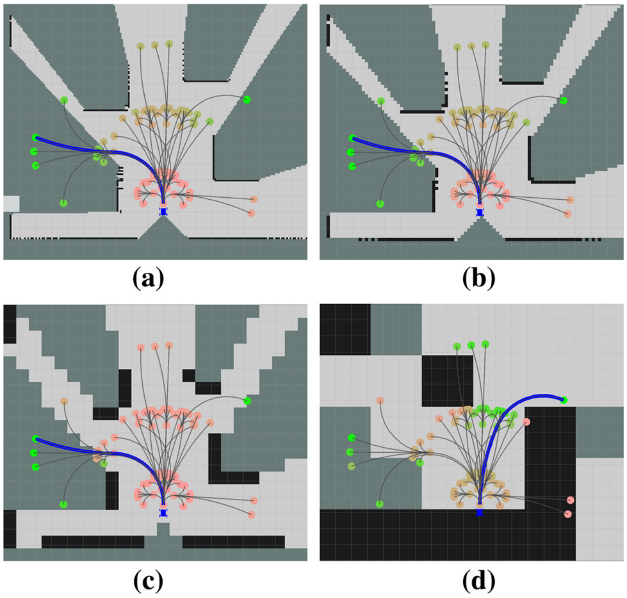


Fig. 10 CSQMI reward along motion primitives computed on compressed OGs. The best exploration path (blue), i.e. that which maximizes (2), is only altered when $n = 5$. Green corresponds to high CSQMI reward, and red to low. **a** m^K, $\Delta = 0.1$ m. **b** $C_1(m^K)$, $\Delta = 0.2$ m , **c** $C_3(m^K)$, $\Delta = 0.8$ m, **d** $C_5(m^K)$, $\Delta = 3.2$ m (Color figure online)

In our ground robot experiments we use $\eta = 0.1$, and $\beta = 0.5$ to persuade larger map compressions (Fig. 6). We set the threshold δ, used to trigger IB optimizations, to 0.05.

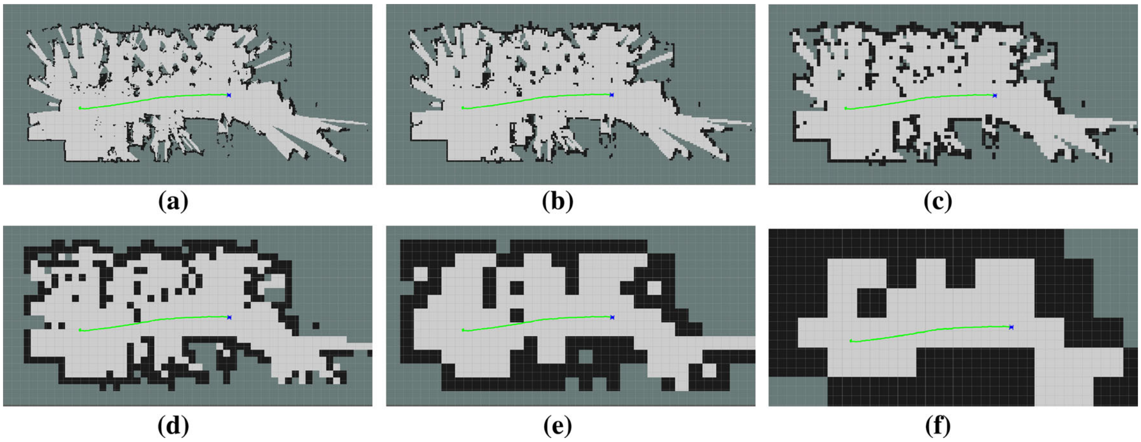


Fig. 11 A six-level 2D OG pyramid, $\mathcal{C}_5(\mathbf{m})$, built from a partially explored base OG, **m**, in a cluttered warehouse environment. The PRI compression strategy preserves boundaries between free and unknown space, and expands occupied cells. Δ denotes cell edge length. The robot is shown in blue, and its path in green. **a** **m**, $\Delta = 0.1$, **b** $C_1(\mathbf{m})$, $\Delta = 0.2$, **c** $C_2(\mathbf{m})$, $\Delta = 0.4$, **d** $C_3(\mathbf{m})$, $\Delta = 0.8$, **e** $C_4(\mathbf{m})$, $\Delta = 1.6$, **f** $C_5(\mathbf{m})$, $\Delta = 3.2$ (Color figure online)

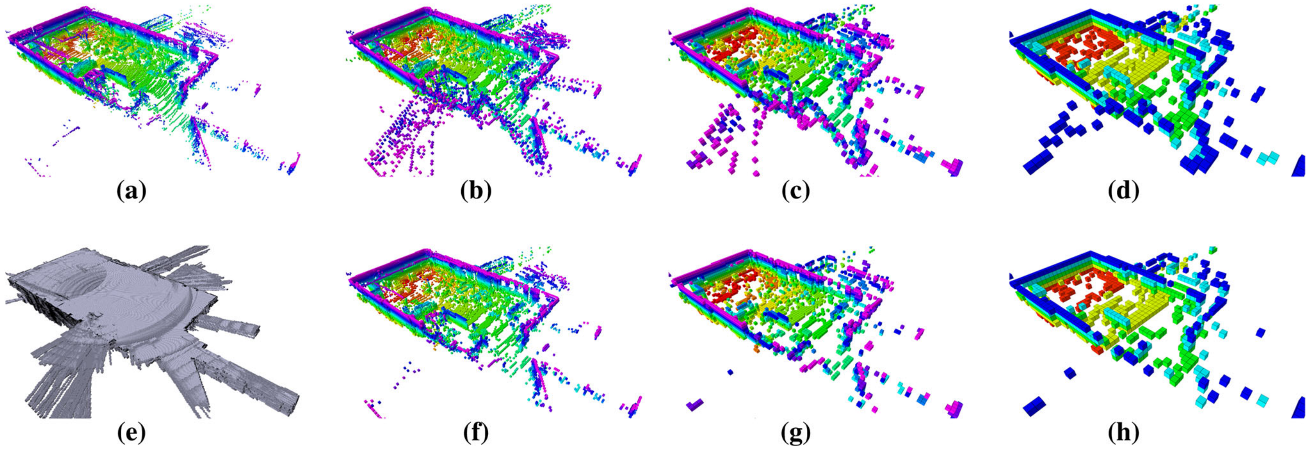


Fig. 12 A four-level 3D OG pyramid, $\mathcal{C}_3(\mathbf{m})$, generated with PRI compression using $\eta = 0.5$ (**b**–**d**) and $\eta = 1$ (**f**–**h**), captured from the same environment as Fig. 11. Only occupied cells are depicted except in (**e**), which shows free (grey) and occupied (black) cells colored by log-odds. **a** Original map, **b** $C_1(\mathbf{m})$, $\Delta = 0.2$, **c** $C_2(\mathbf{m})$, $\Delta = 0.4$, **d** $C_3(\mathbf{m})$, $\Delta = 0.8$, **e** Log-odds coloring, **f** $C_1(\mathbf{m})$, $\Delta = 0.2$, **g** $C_2(\mathbf{m})$, $\Delta = 0.4$, **h** $C_3(\mathbf{m})$, $\Delta = 0.8$ (Color figure online)

This value causes an IB optimization every one to ten hundred meters that the robot travels in the warehouse environment depicted in Figs. 11 and 12. In most environments, compressing a base OG with 0.1 m resolution to 3.2 m resolution ($n > 5$) yields a mostly-free or mostly-occupied compressed OG. We therefore set $n = 5$ as an upper limit when creating OG pyramids.

6.2 Simulation and ground robot experiments

To evaluate the effects of the IB optimization and the PRI compression strategy on the accuracy and efficiency of information-based exploration, we performed experiments in simulation and on a ground robot. In both sets of experiments, we assume that the robot is able to estimate its own state from IMU and laser scan measurements, and produce an accurate OG map of its surroundings in real-time. The assumption of accurate localization and mapping is only valid for feature-rich environments like those shown in our experiments. However, in general one may consider augmenting the exploration cost function with an active localization component (Fox et al. 1998; Stachniss et al. 2004). For state estimation and mapping we use a laser- and inertial-based SLAM implementation similar to the system described by Shen et al. (2011), which leverages ICP for odometry (Pomerleau et al. 2013), a histogram filter for localization, and an unscented Kalman filter (UKF) for sensor fusion (Thrun et al. 2005). We assume no encoder odometry. The OG is updated at 10 Hz and has a 0.1 m resolution. The robot's laser scanner sweeps in a 270° arc with 1081 beams, with a max range of 30 m.

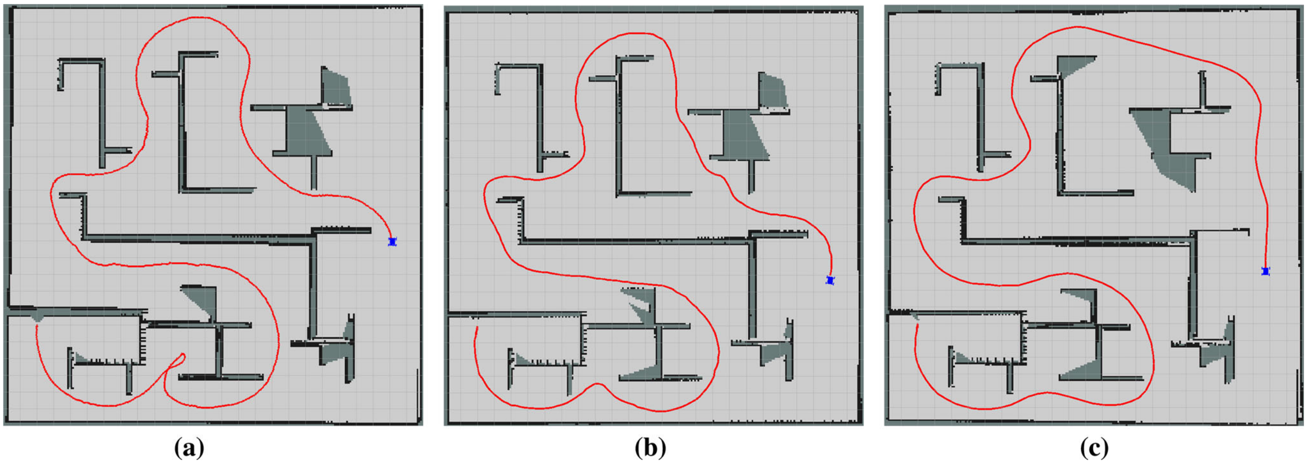


Fig. 13 A robot explores a simulated 25 × 25 m maze environment, planning its actions by evaluating CSQMI against maps with different amounts of compression. The full resolution map with $\Delta = 0.1$ m is depicted in each figure to show map completeness. Varying the compression level n only causes small changes in exploration behavior (e.g. navigating slightly further from walls). However, due to exponential increases to planning frequency, the vehicle is able to increase its maximum velocity from 0.35 m/s to 3.0 m/s (Table 1). **a** $n = 0$, $\Delta = 0.1$ m, $\dot{\mathbf{x}} = 0.35$ m/s, **b** $n = 2$, $\Delta = 0.4$ m, $\dot{\mathbf{x}} = 1.5$ m/s, **c** $n = 4$, $\Delta = 1.6$ m, $\dot{\mathbf{x}} = 3.0$ m/s

Table 1 Simulated exploration trial data (Fig. 13)

n	Δ (m)	Planning freq. (Hz)	Maximum velocity (m/s)	Time (s)
0	0.1	1.5	0.35	230.0
2	0.4	6.0	1.5	54.7
4	1.6	24.0	3.0	31.9

Simulated exploration trials on a maze-like 25 × 25 m map were conducted to examine the effects of OG compression on the robot's path and achievable speed. In each trial, the coarse map used to evaluate CSQMI reward was compressed to a different resolution using (20). Resulting exploration paths are shown in Fig. 13. When calculating CSQMI reward with respect to the original map, the robot is only able to choose actions (2) with a 1.5 Hz frequency before saturating computational resources, leading to a maximum safe velocity of 0.35 m/s. After compressing to $n = 4$, the computational cost of evaluating CSQMI is reduced enough that the robot can plan with a 24.0 Hz frequency, allowing a velocity of 3.0 m/s. As n increases, paths close to walls become occupied in the compressed map and yield low CSQMI reward, so the robot chooses paths in the middle of free space with a higher likelihood. Table 1 shows planning frequency, velocity, and total trial time for these simulations. Exploration was terminated once the CSQMI between each action and the map fell below a threshold value. The action generation strategy used for these trials was myopic, and did not globally consider unexplored areas of the map. Adding a global planning strategy would aid in completing each trial's map.

To evaluate the IB optimization and adaptive strategy introduced in Sect. 5, we explored a 35 × 35 m section of Carnegie Mellon University's Field Robotics Center with a ground robot. The ground robot was equipped with a MicroStrain 3DM-GX3-35 IMU, a Hokuyo URG-30LX 30 m range laser scanner, and an onboard computer with an Intel Core i5 processor and 8 GB RAM. The robot's motor controllers and wheels limit its maximum forward velocity to 1.6 m/s, which was achieved several times throughout the trial. The ground robot, its 72 m exploration path from the environment, and a history of environment model adaptations are depicted in Fig. 14. The amount of compression applied to the map allows the robot to adapt its planning frequency to consider more actions in the same amount of time. Dashed lines in Fig. 14c correspond to moments when the adaptation condition in line 2 of Algorithm 2 is met and colored lines indicate the adaptation condition is met and a new n is computed. Since the planner generates primitives in the robot's forward direction, entropy is generally computed in the local map in front of the robot. n remains at 4 in most of the free regions in the trial, and reduces to 0, 1, and 2 in locations where compression results in large reductions to CSQMI reward (e.g. the first $n = 0$ region occurs as the robot moves through a doorway, the second as it plans to move through a hallway). Planning frequency is modified as n changes, and the robot accelerates or decelerates accordingly.

7 Conclusion and future work

In this work we developed information-theoretic optimizations to reduce the computational expense of active perception planning, enabling a mobile robot to explore its surroundings more efficiently and at higher speeds. We introduced formulations for active perception planning, describ-

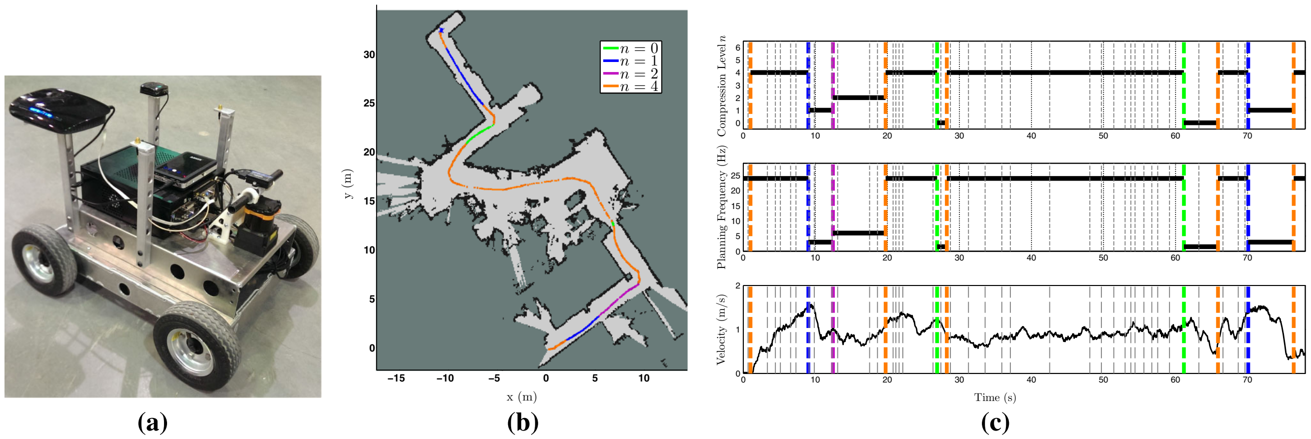


Fig. 14 As the ground robot explores, it recomputes an OG resolution and adapts its planning horizon accordingly. **a** Ground robot, **b** Ground robot exploration path, **c** Adaptation history of n, and its effect on planning frequency and vehicle velocity

ing an information cost function that when optimized gives the action that maximally reduces uncertainty in the robot's representation of its environment. We proposed a method for reducing the computational cost of evaluating this cost function by compressing the robot's environment model to contain only relevant information for exploration, and showed that the IB method can be used to select a compressed environment model that maximizes efficiency of exploration. We then discussed an information-theoretic optimization for generating compressed environment representations using the PRI, and demonstrated a method for adapting the robot's model of its environment in response to its surroundings so as to always keep the best map representation for exploration.

These results were tested through simulation and ground robot experiments, where robots explored previously unknown areas. The presented methods decrease the computational cost of evaluating information-theoretic reward metrics, enabling exploration at higher speeds through cluttered indoor environments.

There are many interesting avenues for future research on the topics of efficient active perception planning as well as environment representations for exploration. First, the IB and PRI optimizations both generalize to other environment representations; future work on this topic could include applying the IB and PRI methods to continuous representations such as Gaussian mixture models of landmarks or Gaussian process maps (Kim and Kim 2012a; T O'Callaghan and Ramos 2012; Kim and Kim 2012b). The idea of reducing the computational cost of active perception planning appears even more valuable in 3D planning scenarios with aerial robots, or for multi-robot planning, where a larger number of actions must be evaluated per planning iteration in order to explore. Finally, it would be interesting to apply the ideas presented to robotic systems capable of achieving more extreme dynamics to examine both the computational and physical factors limiting exploration speed.

References

Amigoni, F., & Caglioti, V. (2010). An information-based exploration strategy for environment mapping with mobile robots. *Robotics and Autonomous Systems*, *58*(5), 684–699.

Basilico, N., Amigoni, F. (2008). On evaluating performance of exploration strategies for autonomous mobile robots. In: IEEE/RSJ *International conference on intelligent robots and systems (IROS), workshop on performance evaluation and benchmarking for intelligent robots and systems, IEEE.*

Bourgault, F., Makarenko, A. A., Williams, S. B., Grocholsky, B., Durrant-Whyte, H. F. (2002). Information based adaptive robotic exploration. In *Intelligent robots and systems, 2002. IEEE/RSJ international conference on, IEEE* (Vol. 1, pp. 540–545).

Burgard, W., Moors, M., Fox, D., Simmons, R., Thrun, S. (2000). Collaborative multi-robot exploration. In *IEEE international conference on robotics and automation (ICRA)* (Vol. 1, pp. 476–481). IEEE

Burt, P. J., & Adelson, E. H. (1983). The laplacian pyramid as a compact image code. *IEEE Transactions on Communication (TCOM)*, *31*(4), 532–540.

Charrow, B. (2015). Information-theoretic active perception for multi-robot teams. Ph.D. thesis, University of Pennsylvania

Charrow, B., Kahn, G., Patil, S., Liu, S., Goldberg, K., Abbeel, P., Michael, N., Kumar, V. (2015a). Information-theoretic planning with trajectory optimization for dense 3D mapping. In *Robotics: Science and system (RSS).*

Charrow, B., Liu, S., Kumar, V., Michael, N. (2015b). Information-theoretic mapping using Cauchy–Schwarz quadratic mutual information. In *IEEE international conference on robotics and automation (ICRA), IEEE.*

Cover, T. M., & Thomas, J. A. (2012). *Elements of information theory*. Hoboken: Wiley.

Einhorn, E., Schroter, C., Gross, H., (2011). Finding the adequate resolution for grid mapping-cell sizes locally adapting on-the-fly. In: *IEEE international conference on robotics and automation (ICRA), IEEE*, pp. 1843–1848.

Elfes, A. (1989). Occupancy grids: A probabilistic framework for robot perception and navigation. Ph.D. thesis, Carnegie Mellon University

Fox, D., Burgard, W., & Thrun, S. (1998). Active markov localization for mobile robots. *Robotics and Autonomous Systems*, *25*(3), 195–207.

Geiger, B. C., Kubin, G. (2013). Signal enhancement as minimization of relevant information loss. In: 9th *International ITG conference on systems, communication, and coding (SCC), VDE*, pp. 1–6.

Holz, D., Basilico, N., Amigoni, F., Behnke, S. (2011). A comparative evaluation of exploration strategies and heuristics to improve them. In *European conference on mobile robots (ECMR),* pp. 25–30

Im, J. J., Leonessa, A., Kurdila, A. (2010). A real-time data compression and occupancy grid map generation for ground-based 3D lidar data using wavelets. In *ASME dynamic systems and control conference (DSCC).*

Julian, B. J. (2013). Mutual information-based gradient-ascent control for distributed robotics. Ph.D. thesis, Massachusetts Institute of Technology.

Julian, B. J., Karaman, S., Rus, D. (2013). On mutual information-based control of range sensing robots for mapping applications. In *IEEE/RSJ international conference on intelligent robots and systems (IROS), IEEE*, pp. 5156–5163.

Kim, S., Kim, J. (2012a). Building large-scale occupancy maps using an infinite mixture of gaussian process experts. In *Australasian conference on robotics and automation (ACRA).*

Kim, S., Kim, J. (2012b). Building occupancy maps with a mixture of gaussian processes. In *IEEE international conference on robotics and automation (ICRA), IEEE*, pp. 4756–4761.

Kollar, T., & Roy, N. (2008). Efficient optimization of information-theoretic exploration in slam. *AAAI Conference on Artificial Intelligence*, *8*, 1369–1375.

Kretzschmar, H., & Stachniss, C. (2012). Information-theoretic compression of pose graphs for laser-based slam. *The International Journal of Robotics Research (IJRR)*, *31*(11), 1219–1230.

LaValle, S. M. (1998). *Rapidly-exploring random trees a new tool for path planning*. Technical report, Compute Science Department, Iowa State University.

Marchant, R., Ramos, F. (2014). Bayesian optimisation for informative continuous path planning. In *2014 IEEE international conference on robotics and automation (ICRA), IEEE*, pp. 6136–6143.

Nelson, E., Michael, N. (2015). Information-theoretic occupancy grid compression for high-speed information-based exploration. In *IEEE/RSJ international conference on intelligent robots and systems (IROS), IEEE.*

O'Callaghan, S. T., & Ramos, F. T. (2012). Gaussian process occupancy maps. *The International Journal of Robotics Research (IJRR)*, *31*(1), 42–62.

Pivtoraiko, M., Kelly, A. (2005). Generating near minimal spanning control sets for constrained motion planning in discrete state spaces. In *IEEE/RSJ international conference on intelligent robots and systems (IROS), IEEE*, pp. 3231–3237.

Pivtoraiko, M., Knepper, R. A., & Kelly, A. (2009). Differentially constrained mobile robot motion planning in state lattices. *Journal of Field Robotics (JFR)*, *26*(3), 308–333.

Pivtoraiko, M., Mellinger, D., Kumar, V. (2013). Incremental micro-uav motion replanning for exploring unknown environments. In *IEEE international conference on robotics and automation (ICRA) Conference on, IEEE*, pp. 2452–2458.

Pomerleau, F., Colas, F., Siegwart, R., & Magnenat, S. (2013). Comparing icp variants on real-world data sets. *Autonomous Robots (AURO)*, *34*(3), 133–148.

Principe, J. C. (2010). *Information theoretic learning: Rényi's entropy and kernel perspectives*. Berlin: Springer.

Richter, C., Ware, J., Roy, N. (2014). High-speed autonomous navigation of unknown environments using learned probabilities of collision. In *Robotics and automation (ICRA), 2014 IEEE international conference on, IEEE*, pp. 6114–6121.

Saarinen, J., Andreasson, H., Stoyanov, T., Ala-Luhtala, J., Lilienthal, A. J. (2013). Normal distributions transform occupancy maps: Application to large-scale online 3D mapping. In IEEE *international conference on robotics and automation (ICRA), IEEE*, pp. 2233–2238.

Shen, S., Michael, N., Kumar, V. (2011). Autonomous multi-floor indoor navigation with a computationally constrained mav. In *IEEE international conference on robotics and automation (ICRA), IEEE*, pp. 20–25.

Shen, S., Michael, N., & Kumar, V. (2012). Stochastic differential equation-based exploration algorithm for autonomous indoor 3D exploration with a micro-aerial vehicle. *The International Journal of Robotics Research (IJRR)*, *31*(12), 1431–1444.

Stachniss, C., Hahnel, D., Burgard, W. (2004). Exploration with active loop-closing for fastslam. In *Intelligent robots and systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on, IEEE*, (Vol. 2, pp. 1505–1510).

Strom, D. P., Nenci, F., Stachniss, C. (2015). Predictive exploration considering previously mapped environments. In *IEEE international conference on robotics and automation (ICRA), IEEE*, pp. 2761–2766.

Taylor, C. J., Kriegman, D. (1993). Exploration strategies for mobile robots. In *Proceedings of robotics and automation, 1993. 1993 IEEE international conference on, IEEE*, pp. 248–253.

Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic robotics*. Boston: MIT press.

Tishby, N., Pereira, F. C., Bialek, W. (2000). The information bottleneck method. arXiv:physics/0004057.

Vallvé, J., Andrade-Cetto, J. (2014). Dense entropy decrease estimation for mobile robot exploration. In IEEE *International conference on robotics and automation (ICRA), IEEE*, pp. 6083–6089.

Wurm, K. M., Hornung, A., Bennewitz, M., Stachniss, C., Burgard, W. (2010). Octomap: A probabilistic, flexible, and compact 3d map representation for robotic systems. In *Proceedings of the ICRA 2010 workshop on best practice in 3D perception and modeling for mobile manipulation*, Vol . 2.

Yamauchi, B. (1997). A frontier-based approach for autonomous exploration. In *IEEE international symposium on computational intelligence in robotics and automation (CIRA), IEEE*, pp. 146–151.

Yoder, L., Scherer, S. (2015). Autonomous exploration for infrastructure modeling with a micro aerial vehicle. In *Field and service robotics (FSR).*

Erik Nelson is a robotics Ph.D. student in the department of Electrical Engineering and Computer Sciences at UC Berkeley. He received an M.S. from the Robotics Institute at Carnegie Mellon University, and a B.S. from the department of Materials Engineering at Cal Poly, SLO. His research brings together robotic perception, exploration, and mapping to enable ground, underwater, and aerial robots to autonomously learn their surrounding environments.

Micah Corah is a Ph.D. student in Robotics Institute at Carnegie Mellon University. Micah received B.S. degrees in Computer Science and Mechanical Engineering from the Rensselaer Polytechnic Institute in 2015. His research interests are in active perception and control for single and multi-robot systems for tasks including exploration and cooperative manipulation with aerial robots.

Nathan Michael is an Assistant Research Professor in the Robotics Institute at Carnegie Mellon University in Pittsburgh, Pennsylvania. His research focuses on enabling autonomous ground and aerial vehicles to robustly operate in uncertain environments with emphasis on robust and adaptive perception, control, and cooperative autonomy.