CARE: Confidence-rich Autonomous Robot Exploration using Bayesian Kernel Inference and Optimization

Yang Xu, *Student Member, IEEE,* Ronghao Zheng[†], *Member, IEEE,* Senlin Zhang, *Member, IEEE,*
Meiqin Liu, *Senior Member, IEEE,* and Shoudong Huang, *Senior Member, IEEE*

Abstract—In this paper, we consider improving the efficiency of information-based autonomous robot exploration in unknown and complex environments. We first utilize Gaussian process (GP) regression to learn a surrogate model to infer the confidence-rich mutual information (CRMI) of querying control actions, then adopt an objective function consisting of predicted CRMI values and prediction uncertainties to conduct Bayesian optimization (BO), i.e., GP-based BO (GPBO). The trade-off between the best action with the highest CRMI value (exploitation) and the action with high prediction variance (exploration) can be realized. To further improve the efficiency of GPBO, we propose a novel lightweight information gain inference method based on Bayesian kernel inference and optimization (BKIO), achieving an approximate logarithmic complexity without the need for training. BKIO can also infer the CRMI and generate the best action using BO with bounded cumulative regret, which ensures its comparable accuracy to GPBO with much higher efficiency. Extensive numerical and real-world experiments show the desired efficiency of our proposed methods without losing exploration performance in different unstructured, cluttered environments. We also provide our open-source implementation code at *https://github.com/Shepherd-Gregory/BKIO-Exploration*.

Index Terms—Mutual information, Bayesian kernel inference, autonomous robot exploration, Bayesian optimization.

Fig. 1. CRMI-based autonomous robot exploration using a range-sensing Fetch robot with limited FOV in a cluttered office environment. **Left**: Fetch robot. **Middle**: Environmental CRMI surface after exploration. It shows the resulting exploration trajectory (white) in one experiment, where the robot's overall exploration sequence to maximize the CRMI is A-B-G-C-D-E-F-D-A-B-G-A. **Right**: Corresponding real-world views at points D and F. The color bar is the MI scale [0, 1] bit. Note that this informative trajectory is planned *online* by the MI-driven planner. We set a high information threshold intentionally to explore rapidly. If setting a lower threshold, the CRMI-driven robot will explore the unexplored spaces (*dark red*), then obscured spaces (*light red*), and be repulsive to walls/obstacles (*deep blue*). The gray lines represent glass rooms, walls, or building boundaries the robot can not enter.

I. INTRODUCTION

ROBOT exploration gains its prevalence recently in *priori* unknown environments such as subterranean, marine, and planetary tasks [1], [2]. Among the literature, state-of-the-art exploration methods prefer to use information-theoretic metrics in each iteration, such as Shannon mutual information (MI) [3] and its derivatives [4]–[7], to evaluate the information gain brought by candidate control actions accurately. For instance, in [3], Julian *et al.* defined the Shannon MI between new observation and the occupancy grid map (OGM) at candidate poses. Charrow *et al.* [4] adopted the Cauchy-Schwarz quadratic MI (CSQMI) to improve the time efficiency
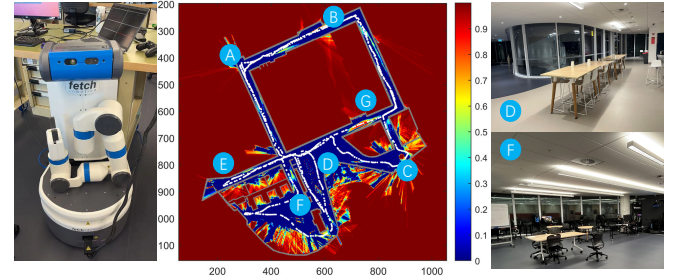
†: Corresponding author.

Yang Xu, Ronghao Zheng, and Senlin Zhang are with the College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China, and also with the State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China. (e-mail: {xuyang94, rzheng, slzhang}@zju.edu.cn)

Meiqin Liu is with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an 710049, China, and also with the State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou 310027, China. (e-mail: liumeiqin@zju.edu.cn)

Shoudong Huang is with the Robotics Institute, Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW 2007, Australia. (e-mail: shoudong.huang@uts.edu.au)

of Shannon MI in 2D and 3D exploration for computationally limited platforms. These OGM-based MI metrics are also called occupancy grid MI (OGMI). Our previous work [6] further provided a more accurate information measure, confidence-rich mutual information (CRMI), over the dense confidence-rich map (CRM) [8] for robot exploration.

Driven by these information metrics, the robot will choose and execute the most informative action, thus the exploration problem becomes a sequential optimal decision-making one naturally. According to [3], the good properties of MI also can ensure the accuracy and the eventual completeness of the exploration. A real-world exploration example is in Fig. 1.

Intuitively, the way to tackle an autonomous exploration problem is to use a greedy strategy and add more candidate actions, including sampled nodes [9], [10], available viewpoints [11], or special motion primitives [12], in the discrete action space. However, the computational cost of the MI evaluation of all candidate actions will become expensive since the forward simulation in the evaluation requires extensive raycasting, map updates, and MI calculation. Notably, these consequences will be more distinct in 3D environments because the increased dimension needs to evaluate much more samples.

A. Related Work

In the context of robot exploration, deep learning has been introduced to realize predicting optimal sensing actions more efficiently. Bai *et al.* [13] trained the deep neural network with plenty of randomly generated 2D gray maps to generate suggested action and ensure inferring in constant time. Graph neural networks have also been combined with reinforcement learning methods to learn the best action from an exploration graph, rather than metric maps or visual images [14], [15]. Nevertheless, the neural network-based robot exploration methods require numerous offline training samples beforehand and are limited to the adaptability and generalization capability in different environments.

More recently, statistical learning techniques also provide a powerful tool to find the global optimum approximately by training predictive models using minor parts of actions in continuous action spaces, without evaluating the objective function expensively, which also has better interpretability in black-box inference [16]–[18]. In [19], Bai *et al.* used the Gaussian process (GP) to model the relationship between control actions and the explicitly evaluated OGMI for the robot exploring *priori* unknown areas. In [20], they further introduced Bayesian optimization (BO) into the OGMI-based robot exploration to optimize the GP prediction in multiple iterations, which provides rapid map entropy reduction and ensures computational efficiency. Generally, BO assesses the acquisition function derived from the GP prior and samples, then chooses the next query point maximizing the acquisition function and balancing the trade-off between exploration (global) and exploitation (local). Iteratively, BO presents more precise results on the posterior distribution as the observations (training samples) increase. Rather than evaluating discrete viewpoints, Francis *et al.* [18] modeled the autonomous exploration and mapping task as a constrained BO aiming to find optimal continuous paths.

However, the main bottleneck of the GP-based methods is that the number of training actions N will affect the resulting prediction accuracy directly, as well as the computational cost. That implies one needs to pay expensive computations to achieve higher exploration performance. Typically, updating and querying the GP models (the engine behind BO) have an overall $\mathcal{O}(N^3)$ time complexity. This compromises the inference efficiency and real-time performance of robot exploration tasks inevitably, especially in larger and 3D scenes.

Notably, the Bayesian kernel inference (BKI) technique proposed in [21] gives us a chance to perform efficient exact inference on a simplified model, rather than approximating inference on an exact generative model (e.g. GP) expensively. BKI extends local kernel estimation to Bayesian inference for exponential likelihood functions, enabling only $\mathcal{O}(\log N_q)$ (N_q: the number of querying samples) run time for inference. These significant merits enhance BKI's application in robotics, including sensor uncertainty estimation [22], high-speed navigation [23], as well as environment mapping using sparse sensor measurements such as terrain traversability mapping [24], 3D occupancy mapping [25], semantic mapping [26].

B. Motivations and Contributions

Motivated by the recent BKI technique [21] and its robotic applications [24], [25], [27], in this paper, we aim to utilize it to accelerate the computationally expensive CRMI evaluation of control actions considerably without losing CRMI prediction accuracy and exploration performance in complex scenes and use BO to optimize the decision-making process. Our main contributions are three-fold:

1) We first present a regular CRMI inference method based on GP and BO (GPBO) using minor explicitly evaluated samples to predict CRMI values and uncertainties of candidate actions, and use an information-theoretic objective function to realize the trade-off of the exploration-exploitation dilemma in MI prediction, rather than the naive-greedy (NG) method evaluating all candidate actions exhaustively and expensively;

2) We further propose a novel Bayesian kernel inference and optimization method (BKIO) for CRMI inference without training a surrogate evaluation model. Compared with the GPBO method, BKIO can perform much more efficient MI values and uncertainties prediction in approximate logarithm time with comparable accuracy and exploration performance;

3) We conduct extensive numerical simulations in synthetic and dataset maps, and real-world experiments in a cluttered office scene to validate our proposed methods. We also release an open-source implementation of our proposed methods[1].

The paper's organization is as follows. We formulate the GPBO-based CRMI inference problem in Section II and present BKIO for CRMI in Section III. Simulation results using synthetic data and the real-world experimental results, as well as the discussion, are given in Section IV, followed by conclusions in Section V.

II. BO-BASED CRMI-DRIVEN ROBOT EXPLORATION

In this paper, for simplicity of discussion, we mainly consider the CRMI-based information-theoretic exploration using a mobile robot equipped with a beam-based range sensor of a limited field of view (FOV) in 2D environments. We aim to evaluate the exact CRMI of unknown robot configurations sampled in the action space efficiently.

A. Information-Theoretic Exploration using CRMI

Considering the information-based autonomous robot exploration in an unknown, cluttered, and unstructured environment, we prefer to use CRMI as the information measure, since CRMI is more descriptive and accurate in cluttered and unstructured areas than the OGMI derived from Bernoulli distribution-based OGMs, especially when using sparse measurements or low-resolution sensors [6]. This advantage stems from the underlying CRM, a dense environmental representation consisting of non-parametric occupancy distribution over map cells [8], which can encode the measurement dependencies between map cells within a sensor cone into the map belief over each cell, thereby enhancing the perception ability of the robot in uncertainty-aware navigation and planning tasks.

Here we begin with some related definitions and notions. Generally, the robot generates a set of candidate actions $\mathcal{X}_{\text{action}}$ in the robot's feasible configuration space $\mathcal{X} \subseteq SE(2)$. For a candidate configuration $x_i = [p_i^x, p_i^y, \psi_i] \in \mathcal{X}_{\text{action}}$, p_i^x and p_i^y denote the robot's position on the map, and ψ_i

---

[1] https://github.com/Shepherd-Gregory/BKIO-Exploration

denotes the heading angle of the robot. We assume this configuration space has been discretized by a fixed resolution over the 2D static grid map comprised of N_m grid cells. The continuous occupancy values over this map are $m = \{m^{[1]}, \ldots, m^{[\xi]}, \ldots, m^{[N_m]}\}(m^{[\xi]} \in [0,1])$, i.e. the occupancy level over the independent grid cells, which can be updated and queried by the CRM method.

In a typical CRM, the map belief $b_t^{m^{[\xi]}} := p(m^{[\xi]}|z_{1:t}, x_{1:t})$ of a cell $m^{[\xi]}$ at time t is a non-parametric continuous occupancy distribution conditioned upon the sensor observations $z_{1:t}$ and robot poses $x_{1:t}$. The expected map occupancy over a cell $m^{[\xi]}$ can be defined as the mathematical expectation of the map belief $b^{m^{[\xi]}}$: $\hat{m}^{[\xi]} := \mathbb{E}[m^{[\xi]}] = \int_0^1 m^{[\xi]} b^{m^{[\xi]}} dm^{[\xi]}$, and the confidence of map occupancy can be defined as the variance of map belief distribution. The occupancy distribution of an unobserved map cell $m^{[\xi]}$ is assumed to be uniform, i.e., $\hat{m}^{[\xi]} = 0.5$. This unobserved cell contributes an entropy of 1 bit and a totally known cell (i.e. $\hat{m}^{[\xi]} = 0$ or 1) contains 0-bit entropy according to the information theory.

From the view of information theory, the expected information gain of x_i can be evaluated by the current map entropy and conditional entropy given a new measurement at x_i:

$$I(m; x_i) = H(m) - H(m|x_i). \quad (1)$$

The aim of information-theoretic robot exploration is to select the best action x_{best} maximizing the expected CRMI:

$$x_{\text{best}} = \operatorname*{argmax}_{x_i \in \mathcal{X}_{\text{action}}} I(m; x_i). \quad (2)$$

Generally, the expected CRMI of a candidate configuration x_i can be calculated by the *virtual* measurement of a scan z' raycasting on currently built map m at pose x_i, then updating a virtual map m' and decomposing the CRMI of a scan into each cell on each independent beam, thus the CRMI can be accumulated over all cells on the map approximately, so Eq. (1) can be rewritten as follows:

$$I(m; x_i) = H(m) - H(m'|x_i, z')$$
$$\approx \sum_{j=1}^{N_z} \sum_{k \in \mathcal{Z}_{[j]}} I(m'_{[k]}; z'_{[j]}), \quad (3)$$

where N_z is the number of beams of the virtual scan z', $\mathcal{Z}_{[j]}$ is the index set of map cells intersected with the jth beam $z'_{[j]}$ of scan z'.

The CRMI of the cell $m'_{[k]}$ with the measurement beam $z'_{[j]}$ can be calculated by the following equation using numerical integration:

$$I(m'_{[k]}; z'_{[j]}) = H(m'_{[k]}|z_{1:t-1}) - H(m'_{[k]}|z'_{[j]}, z_{1:t-1})$$
$$= \int_{z \in Z_m} p(z'_{[j]}|z_{1:t-1}) \int_0^1 b_t^{m'_{[k]}} \log b_t^{m'_{[k]}} dm'_{[k]} dz'_{[j]}$$
$$- \int_0^1 b_{t-1}^{m'_{[k]}} \log b_{t-1}^{m'_{[k]}} dm'_{[k]}, \quad (4)$$

where Z_m is the maximum measurement range. For more details about CRMI calculation, please see [6].

Proposition 1 *[6] The CRMI calculation process involves numerous numerical integration and accumulation operations.*

Hence, the overall computational complexity of evaluating a configuration is about $O(N_z N_c^2)$ for updating the map and $O(N_z N_c^2 \lambda_z^{-1} \lambda_m^{-1})$ for the MI calculation, where λ_z and λ_m are the numerical integration resolutions in Eq. (4), N_c is the maximum size of $\mathcal{Z}_{[j]}$.

The explicit CRMI evaluation of all sampled actions imposes notable computational costs for robot exploration, especially in unstructured, cluttered, and 3D scenes, since the robot needs to assess more actions to keep safety and exploration performance in these cases.

B. BO-based CRMI Inference for Robot Exploration

To alleviate the computational burden of evaluating numerous actions, here we first introduce GP and BO to infer the CRMI with a training set of explicitly evaluated actions.

Consider a supervised learning-based inference problem on predictive stochastic models $p(\mathbf{y}|\mathbf{x})$ given a sequence of N explicitly evaluated samples $\mathcal{D} = \{(\mathbf{x} = \{x_i\}, \mathbf{y} = \{y_i\})\}_{i=1}^N$, where $\mathbf{x}$ and $\mathbf{y}$ represent the set of evaluated configurations and the resulting CRMI values $I(m; \mathbf{x})$, respectively. The main objective is to infer the posterior distribution $p(\mathbf{y}^*|\mathbf{x}^*, \mathcal{D})$ to evaluate CRMI values $\mathbf{y}^* \in \mathbb{R}^{N_q}$ of the N_q querying sample inputs $\mathbf{x}^* \in \mathcal{X}_{\text{action}}$.

Here we assume $p(\mathbf{y}|\mathbf{x})$ follows a GP, according to [28], we can estimate the posterior mean $\bar{\mathbf{y}}^*$ and covariance $cov(\mathbf{y}^*)$ of the output of the querying set of candidate actions $\mathbf{x}^*$, i.e.:

$$\bar{\mathbf{y}}^* = k(\mathbf{x}^*, \mathbf{x})[k(\mathbf{x}, \mathbf{x}) + \boldsymbol{\sigma}_N^2 \mathbf{I}]^{-1} \mathbf{y}, \quad (5)$$
$$cov(\mathbf{y}^*) = k(\mathbf{x}^*, \mathbf{x}^*) - k(\mathbf{x}^*, \mathbf{x})[k(\mathbf{x}, \mathbf{x}) + \boldsymbol{\sigma}_N^2 \mathbf{I}]^{-1} k(\mathbf{x}, \mathbf{x}^*),$$

where $\boldsymbol{\sigma}_N^2$ is the Gaussian noise variance vector associated with input $\mathbf{x}$, $k(\cdot, \cdot)$ is a kernel function measuring the proximity between two elements in feature space.

Among the available kernel functions, we prefer Matérn kernel for its capability of handling sudden transitions of terrain [28], [29] since the potential obstacles and unknown structures in application scenes that have never been seen before will vary the MI values greatly. In practice, we choose a Matérn 5/2 kernel with the form as:

$$k(\mathbf{x}^*, \mathbf{x}) = (1 + \frac{\sqrt{5}r}{\ell} + \frac{5r^2}{3\ell^2}) \exp(-\frac{\sqrt{5}r}{\ell}), r = ||\mathbf{x}^* - \mathbf{x}||, \quad (6)$$

where ℓ is a positive characteristic length scale.

At each exploration step, the information-driven robot expects to choose the actions with high predicted CRMI values (*exploitation*) to maximize the information gain locally (i.e. Eq. (2)). Still, the results directly predicted by GP tend to recommend sub-optimal action because of the insufficient evaluated samples and the resulting inaccurate predictive model. Hence, the actions with high predicted uncertainty can also not be neglected in the inference process (*exploration*).

To handle this, we incorporate the prediction uncertainty of CRMI values to the objective function in Eq. (2) to realize a local trade-off between exploration and exploitation. Referring to the GP-UCB in [30], here we can get the suggested action maximizing the information objective function f^I based on Eq. (2) and Eq. (5) after multiple epochs of optimization:

$$x_s = \operatorname*{argmax}_{\mathbf{x} \in \mathcal{X}_{\text{action}}} f^I, f^I = \bar{\mathbf{y}} + \alpha^{1/2} \sigma_{\mathbf{y}}, \quad (7)$$

where α is the trade-off factor, $\sigma_{\mathbf{y}}$ is the standard deviation of CRMI prediction.

For such a sequential multi-armed bandit problem [31], one can use the expected regret to ensure that the multi-epoch optimization is reasonable and practical, i.e. the regret of the upper confidence bound (UCB) function should be bounded over rounds.

Lemma 1 *[30] For finite $|\mathcal{D}| < \infty$, let $\delta \in (0,1)$ and $\alpha = 2\log(|\mathcal{D}|T^2\pi^2/6\delta)$, the UCB with α for a sample of a GP with mean function zero and covariance function $k(\mathbf{x}^*, \mathbf{x})$, can obtain a regret bound of $\mathcal{O}^*(\sqrt{N_{\text{epoch}}\gamma_T \log|\mathcal{D}|})$ with high probability, where N_{epoch} is the optimization rounds, $T \in [1, N_{\text{epoch}}]$ is the current optimization round, and γ_T is the max information gain after N_{epoch} rounds.*

Remark 1 *According to Lemma 1, the GP-based BO using the UCB of Eq. (7) and the Matérn 5/2 kernel can achieve the desired trade-off of exploration and exploitation after N_{epoch} rounds optimization.*

III. BAYESIAN KERNEL INFERENCE AND OPTIMIZATION FOR CRMI-BASED ROBOT EXPLORATION

In this section, we present a novel and highly efficient CRMI inference method based on the BKI and BO for autonomous robot exploration.

A. Bayesian Generalized Kernel Inference

Here we continue to use the dataset $\mathcal{D}$ and the querying set $\mathbf{x}^*$ in Section II-B, and assume the CRMI output $\mathbf{y}^*$ and the actions input $\mathbf{x}^*$ follows the relationship of $p(\mathbf{y}|\mathbf{x})$ (not a GP model). According to the BKI theory and applications [21], [24], [32], the Bayesian generalized kernel inference problem can be formulated into three steps:

1) *Target parameters inference*: First, this problem can be solved by associating latent parameters $\boldsymbol{\theta} = \{\theta_i\}_{i=1}^N \in \boldsymbol{\Theta}$ with input $\mathbf{x}$ in the latent space $\boldsymbol{\Theta}$, where the likelihood $p(\mathbf{y}|\boldsymbol{\theta})$ is known. Thus the inference on $\mathbf{y}^*$ can be formulated as an inference on target parameters $\boldsymbol{\theta}^* \in \boldsymbol{\Theta}$ related to $\mathbf{x}^*$:

$$p(\mathbf{y}^*|\mathbf{x}^*, \mathcal{D}) = \int_{\boldsymbol{\Theta}} p(\mathbf{y}^*|\boldsymbol{\theta}^*)p(\boldsymbol{\theta}^*|\mathbf{x}^*, \mathcal{D})d\boldsymbol{\theta}^*, \quad (8)$$

where the posterior distribution of the latent variables can be characterized using Bayes' rule: $p(\boldsymbol{\theta}^*|\mathbf{x}^*, \mathcal{D}) \propto \int_{\boldsymbol{\Theta}} \prod_{i=1}^N p(y_i|\theta_i)p(\boldsymbol{\theta}_{1:N}, \boldsymbol{\theta}^*|\mathbf{x}_{1:N}, \mathbf{x}^*)d\boldsymbol{\theta}_{1:N}$.

2) *Latent variables marginalization*: By strongly assuming latent parameters $\boldsymbol{\theta}_{1:N}$ are conditionally independent given the target parameters $\boldsymbol{\theta}^*$: $p(\boldsymbol{\theta}_{1:N}, \boldsymbol{\theta}^*|\mathbf{x}_{1:N}, \mathbf{x}^*) = \prod_{i=1}^N p(\theta_i|\boldsymbol{\theta}^*, x_i, \mathbf{x}^*)p(\boldsymbol{\theta}^*|\mathbf{x}^*)$, one can marginalize the latent variables $\boldsymbol{\theta}_{1:N}$ and then obtain $p(\boldsymbol{\theta}^*|\mathbf{x}^*, \mathcal{D}) \propto \prod_{i=1}^N p(y_i|\boldsymbol{\theta}^*, x_i, \mathbf{x}^*)p(\boldsymbol{\theta}^*|\mathbf{x}^*)$.

3) *Kernel approximation*: BKI further defines a distribution that has a special smoothness constraint and bounded Kullback-Leibler divergence $D_{KL}(g||f)$ between the extended likelihood $p(y_i|\boldsymbol{\theta}^*, x_i, \mathbf{x}^*)$ represented by g and the likelihood $p(y_i|\theta_i)$ represented by f, i.e., the maximum entropy distribution g satisfying $D_{KL}(g||f) \leq \rho(\mathbf{x}^*, \mathbf{x})$ has the form $g(\mathbf{y}) \propto f(\mathbf{y})^{k(\mathbf{x}^*, \mathbf{x})}$, where $\rho(\cdot, \cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^+$ is a smoothness bound and $k(\cdot, \cdot) : \mathcal{X} \times \mathcal{X} \to [0, 1]$ is the kernel

function which can be uniquely determined by ρ. Substituting it into Eq. (8) yields:

$$p(\boldsymbol{\theta}^*|\mathbf{x}^*, \mathcal{D}) \propto \prod_{i=1}^N p(y_i|\boldsymbol{\theta}^*)^{k(\mathbf{x}^*, \mathbf{x})}p(\boldsymbol{\theta}^*|\mathbf{x}^*). \quad (9)$$

Thus, the posterior distribution can be exactly inferred by using the likelihood from the exponential family (e.g. Gaussian distribution) and assuming the corresponding conjugate prior.

B. BKI Optimization for Autonomous Robot Exploration

Here we present the BKIO method for information-driven robot exploration, using BO to optimize BKI results in multiple epochs.

1) *Bayesian kernel CRMI inference*: According to Section III-A, we assume the underlying likelihood $p(\mathbf{y}|\boldsymbol{\theta})$ follows a Gaussian distribution with an unknown mean vector $\boldsymbol{\theta}$ and a fixed, known covariance matrix function $\boldsymbol{\Sigma}(\mathbf{x})$:

$$\mathbf{y} \sim \mathcal{N}(\boldsymbol{\theta}, \boldsymbol{\Sigma}(\mathbf{x})), \boldsymbol{\Sigma}(\mathbf{x}) = \text{diag}(\sigma_\theta^2) \in \mathbb{R}^{N \times N}, \quad (10)$$

where σ_θ^2 is the known variance associated with $\boldsymbol{\theta}$. Thus the conjugate prior of $p(\mathbf{y}|\boldsymbol{\theta})$ can also be described by a Gaussian distribution using the hyperparameter ζ and samples input $\mathbf{x}$:

$$p(\boldsymbol{\theta}|\mathbf{x}) = \mathcal{N}(\boldsymbol{\theta}_0, \boldsymbol{\Sigma}(\mathbf{x})/\zeta), \quad (11)$$

where $\boldsymbol{\theta}_0$ and ζ are the initial belief of the mean and the uncertainty of that belief, respectively. $\boldsymbol{\theta}_0$ is a vector consisting of N known equal constants θ_0, and $\zeta = 0$ means no confidence and $\zeta \to \infty$ indicates full prior knowledge. Here we assume ζ is a quite small positive constant since we do not have much prior information about the belief when exploring unknown areas. Therefore, since the querying samples set $\mathcal{D}^* = \{(\mathbf{x}^* = \{x_i^*\}, \mathbf{y}^* = \{y_i^*\})\}_{i=1}^{N_q}$ also follows the distributions of Eq. (10) and Eq. (11), given the set of explicitly evaluated samples $\mathcal{D}$, substituting Eq. (11) and Eq. (10) into Eq. (9) yields:

$$p(\boldsymbol{\theta}^*|\mathbf{x}^*, \mathcal{D}) \propto \prod_{i=1}^N \exp\left(-\frac{(y_i - \theta_i)^2}{2\sigma_\theta^2}k(\mathbf{x}^*, \mathbf{1}_{N_q} \times x_i)\right) \quad (12)$$
$$\cdot \exp\left(-\frac{1}{2}\frac{(\theta_i - \theta_0)^2}{\sigma_\theta^2}\zeta\right),$$

thus the predicted mean and covariance of the CRMI posterior can be derived as follows:

$$\bar{\mathbf{y}}^* = \mathbb{E}[\mathbf{y}^*|\mathbf{x}^*, \mathcal{D}] = \mathbb{E}[\boldsymbol{\theta}^*|\mathbf{x}^*, \mathcal{D}] = \frac{\bar{\mathbf{y}} + \zeta\theta_0}{\zeta + \bar{k}} \simeq \frac{\bar{\mathbf{y}}}{\bar{k}},$$

$$cov(\mathbf{y}^*) = \mathbb{V}[\boldsymbol{\theta}^*|\mathbf{x}^*, \mathcal{D}] = \frac{\boldsymbol{\Sigma}(\mathbf{x}^*)}{\zeta + \bar{k}} \simeq \frac{\boldsymbol{\Sigma}(\mathbf{x}^*)}{\bar{k}}, \quad (13)$$

where $\bar{\mathbf{y}}$ and $\bar{k}$ can be computed by the kernel function:

$$\bar{k} = \Sigma_{i=1}^N k(\mathbf{x}^*, \mathbf{x}), \ \bar{\mathbf{y}} = \Sigma_{i=1}^N k(\mathbf{x}^*, \mathbf{x})y_i. \quad (14)$$

Therefore, given a set $\mathcal{D}$ of explicitly evaluated samples as the input dataset, we can easily compute the CRMI and the corresponding prediction uncertainty for the querying configurations $\mathbf{x}^*$ by using Eq. (13) and Eq. (14).

2) *BKI optimization*: To optimize the BKI inference results of CRMI, we apply Bayesian optimization and use the

objective function Eq. (7) to ensure the trade-off between exploration and exploitation at each step.

Remark 2 *According to [30], as a special finite case of GP, the multivariate Gaussian posterior over $\mathbf{y}^*$ inferred by BKIO (c.f. Eq. (13)) can also ensure the Bayesian optimization regret is bounded using the Matérn 5/2 kernel function and the UCB function of f^I in Eq. (7), which improves the optimality of final suggested action after multiple epochs optimization in each exploration step.*

The autonomous exploration framework based on the BKIO method is given in Algorithm 1, where the function 'GenActions' (Line 5) generates $N + N_q$ actions uniformly in the robot's current sensing range, 'ComputeCRMI' (Line 12) refers to Eq. (3) and Eq. (4), 'BKIOptimization' (Line 16) is in the Algorithm 2. In each iteration, the robot first generates feasible actions x_{cand} in its sensing range, e.g., using frontier-based or sampling-based methods, then computes the CRMI values of the actions $\mathbf{x}$ randomly sampled from x_{cand} (Lines 5-14). The robot uses Algorithm 2 to obtain the sets of best actions and the associated MI values and chooses the action with the maximum MI if it's higher than the threshold, otherwise, the robot will go back to the last action to end the expansion in this branch (Lines 16-23). Once the best action is confirmed, the robot will use the classic A* algorithm to plan the local collision-free path from the current pose to the best action, and then update the global map using the measurements and poses along the planned local path (Lines 25-26). The exploration will end when the iteration counts meet N_{loop} or the action history set is empty.

In Algorithm 2, for each optimization epoch, we first compute the kernel function matrix and use it to compute the CRMI values with the covariance of the querying set (Lines 4-7). Then we can obtain the objective function values of all querying actions and the action with the highest objective function value (Lines 8-9). We can add it into the best action set if it is already in $\mathcal{D}$; otherwise, we will evaluate it explicitly using Eq. (3) and add it into $\mathcal{D}$ to enlarge the reference dataset and enhance the prediction accuracy (Lines 10-18).

Proposition 2 *The time complexity of our proposed BKIO method for evaluating all generated actions in Algorithm 1 and 2 is $\mathcal{O}(N(N_z N_c^2 + N_z N_c^2 \lambda_z^{-1} \lambda_m^{-1}))$ of explicit CRMI evaluation and $\mathcal{O}(N_{\text{epoch}} N \log N_q)$ of BKIO. The GPBO-based CRMI exploration methods in Section II-B have the same time complexity of explicit CRMI evaluation as BKIO but a complexity of $\mathcal{O}(N_{\text{epoch}}(N^3 + N^2 N_q))$ to perform the expensive GP inference for CRMI, where N_{epoch} is the number of optimization epochs.*

Comparatively, the NG-based exploration method which evaluates all actions explicitly has a time complexity of $\mathcal{O}((N + N_q)(N_z N_c^2 + N_z N_c^2 \lambda_z^{-1} \lambda_m^{-1}))$ for explicit CRMI evaluation according to **Proposition** 1. The above comparative results indicate that the BKIO-based exploration method significantly outperforms GPBO and NG methods in time efficiency, especially in large-scale and cluttered places that need more samples to evaluate rapidly (much greater N and N_q).

Algorithm 1 BKIO-Exploration( )

Require: Occupancy map at kth time step m_k, previous robot poses $x_{\text{hist}} = x_{0:k-1}$ and current pose x_k, the number of explicitly evaluated samples N, information threshold I_{th}, the number of querying samples N_q, loop limit N_{loop}

1: iter = 0
2: **while** $x_{\text{hist}} \neq \emptyset$ OR iter $< N_{\text{loop}}$ **do**
3:     iter = iter + 1;
4:     *// Generate $N + N_q$ actions uniformly near x_k*
5:     $\mathbf{x}_{\text{cand}} \leftarrow$ **GenActions**(x_k, m_k);
6:     *// Sample N actions randomly from $\mathbf{x}_{\text{cand}}$*
7:     $\mathbf{x} \leftarrow$ **RandSample**$(\mathbf{x}_{\text{cand}}, N)$;
8:     $\mathbf{x}^* \leftarrow \mathbf{x}_{\text{cand}} \setminus \mathbf{x}$; *// The set of N_q querying actions*
9:     *// Evaluate these reference actions explicitly*
10:    **for** each $x_i \in \mathbf{x}$ **do**
11:       $m_{\text{vir}} \leftarrow$ **Raycasting**(x_i, m_k);
12:       $I_i \leftarrow$ **ComputeCRMI**(m_{vir}, x_i);
13:       $\mathbf{y} \leftarrow \mathbf{y} \cup I_i$;
14:    **end for**
15:    *// Find the suggested action using Algorithm 2*
16:    $\{x_{\text{best}}, I_{\text{best}}\} \leftarrow$ **BKIOptimization**$(\{\mathbf{x}, \mathbf{y}\}, \mathbf{x}^*, m_k)$;
17:    **if** **Max**$(I_{\text{best}}) > I_{\text{th}}$ **then**
18:       $x_{k+1} \leftarrow x_{\text{best}}(\text{MaxInfoIndex})$;
19:       $x_{\text{hist}} \leftarrow x_{\text{hist}} \cup x_{k+1}$;
20:    **else**
21:       $x_{k+1} \leftarrow x_{k-1}$; *// Back to previous action*
22:       Remove x_{k-1} from x_{hist};
23:    **end if**
24:    *// Execute the action and update the map*
25:    $\mathcal{P}_{\text{local}} \leftarrow$ **Astar**(x_k, x_{k+1}) *// Plan local path by A**
26:    $m_{k+1} \leftarrow$ **ConfidenceRichMapping**$(\mathcal{P}_{\text{local}})$;
27: **end while**

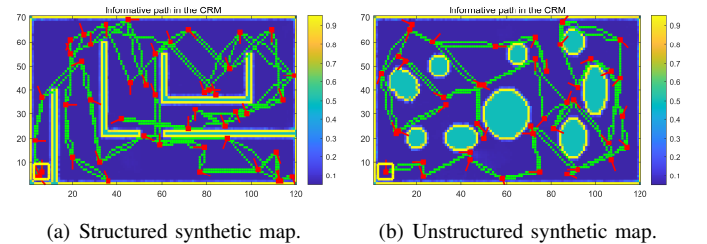(a) Structured synthetic map.   (b) Unstructured synthetic map.

Fig. 2. Examples of CRMI-based robot exploration in unknown synthetic environments. *Yellow square*: the start and end points. The red line starting with a red square represents a planned robot action/pose in SE(2), and *green square* is the local path between two actions.

IV. RESULTS AND DISCUSSIONS

In this section, we aim to validate the performance of the proposed BKIO method, compared with other famous exploration methods such as naive greedy-based ("NG") [6], and GPBO with multiple epochs ("GPBO"), where GPBO replaces Line 16 of Algorithm 1 with the function 'GPBOptimization' using Eq. (5). NG evaluates all actions ($\mathbf{x}$ and $\mathbf{x}^*$) explicitly using Algorithm 1 (except Lines 15-16). Note that **NG** in [6] can be treated as the *ground truth* for verifying the CRMI prediction accuracy and exploration performance.

Algorithm 2 BKIOptimization( )

Require: Occupancy map at kth time step m_k, the set of explicitly evaluated samples $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, the set of querying actions $\mathbf{x}^*$, the number of optimization epoch N_{epoch}, factor α
1: $x_{\text{best}} \leftarrow \{\}, f_{\text{best}}^{I} \leftarrow \{\}$;
2: **for** each epoch **do**
3:     // *Compute the kernel function using* Eq. (6)
4:     $k \leftarrow$ **KernelFunction**$(\mathbf{x}^*, \mathbf{x})$;
5:     // *Infer CRMI and uncertainty using* Eq. (13)
6:     $\overline{k} \leftarrow \Sigma k, \ \overline{y} \leftarrow k \cdot \mathbf{y}$;
7:     $\mathbf{y}^* \leftarrow \overline{y}/\overline{k}, \ cov(\mathbf{y}^*) \leftarrow \Sigma/\overline{k}$;
8:     $f^{I} \leftarrow \mathbf{y}^* + \alpha^{1/2}\sigma_{\mathbf{y}^*}$;
9:     $[x_s, \text{idx}] \leftarrow$ **Max**$(\{f^{I}, \mathbf{y}\})$;
10:     **if** $x_s \notin \mathbf{x}$ **then**
11:         // *Evaluate the CRMI of x_s explicitly*
12:         $m_{\text{vir}} \leftarrow$ **Raycasting**(x_s, m_k);
13:         $y_s =$ **ComputeCRMI**(m_{vir}, x_s);
14:         // *Add into $\mathcal{D}$ to improve prediction accuracy*
15:         $\mathbf{x} \leftarrow \mathbf{x} \cup x_s, \mathbf{y} \leftarrow \mathbf{y} \cup y_s, \mathbf{x}^* \leftarrow \mathbf{x}^* \setminus x_s$;
16:     **else**
17:         $y_s \leftarrow \mathbf{y}(\text{idx})$;
18:     **end if**
19:     // *Save actions with highest objective function values*
20:     $x_{\text{best}} \leftarrow x_{\text{best}} \cup x_s, I_{\text{best}} \leftarrow I_{\text{best}} \cup y_s$
21: **end for**
22: **return** $x_{\text{best}}, I_{\text{best}}$

A. Synthetic Environments and Dataset Results

We run 20 Monte Carlo (MC) trials for all numerical simulations and dataset experiments on a laptop PC with a 2.6 GHz Intel i5-4210M CPU and 8G RAM. The robot poses are obtained by Gmapping [33]. Note that the man-made binary ground truth maps are used only to simulate the true world and generate virtual sensor observations for robot localization and mapping, and the robot cannot access these maps for exploration purposes, i.e., it still faces *unknown environments*.

The minimum information threshold is $I_{\text{th}} = 2$ bits, and the trade-off factor is $\alpha = 1$. The λ_z and λ_m for CRMI calculation are both 0.1. We also choose the parameters of $\zeta = 0.001$ and $\sigma_\theta = 100$ for the BKIO method. We use 30 points (each point with eight directions uniformly) to generate candidate actions for each method, i.e., $N + N_q = 240, N_q = 2N$. We also set $N_{\text{epoch}} = 30$ for GPBO and BKIO methods. This value is chosen particularly and validated in an experimental study. More related details will be particularly analyzed below. Note that we set an extreme case of numerous candidate actions to test the time-saving performance of the proposed methods.

To compare the exploration performance using different methods more quantitatively, referring to [4], [5], we prefer to use map entropy and coverage rate of MC results since there exists randomness in the exploration process. In the map entropy and coverage rate figures (c.f. Fig. 3 and Fig. 6), the solid lines depict the means of MC trials of each method, and the shaded regions represent the standard deviations. Here we omit the standard deviation of NG for the figure's readability.

Results from structured/unstructured cases: To simulate indoor and field scenes, we generate two synthetic maps with

the size of 24 m × 14 m, one structured maze map with several walls (Fig. 2(a), $N_{\text{loop}} = 100$), and one unstructured map consisting of circles and ellipses representing forest-like scene (Fig. 2(b), $N_{\text{loop}} = 150$). The unstructured one will obstruct the robot' FOV and need more exploration steps. The map resolutions are both 0.2 m. The simulated range sensor has a FOV of ± 1.5 rad with a resolution of 0.05 rad and a maximum sensing range of 6 m. The robot is initially at [1.2, 1.2] m with 0 rad heading and trying to explore the prior unknown map.

The examples of resulting informative paths maximizing CRMI values are shown in Fig. 2. The quantitative results of structured and unstructured maps are shown in Fig. 3. BKIO and GPBO have similar evolution trends of map entropy reduction and coverage rate as NG in structured and unstructured maps, indicating the relatively low exploration performance loss in these BO-based methods.

Results from dataset case: To test our methods in a more realistic environment, we use a 24 m × 14 m Seattle map [34] containing narrow long corridors and cluttered rooms in Fig. 5(a). To implement the exploration methods, we use the same simulated laser scanner as in the synthetic maps. The initial robot pose is $x_0 = [11.4, 2.6, -\pi/2]$. Fig. 5 (a) selects one resulting informative path in Monte Carlo experiments. In the Seattle map, Fig. 6 shows BKIO and GPBO methods have similar reduction rates of map entropy and coverage rates w.r.t exploration steps. The simulation results provide evidence BKIO and GPBO can achieve comparable exploration performance in different environments.

(a) Map entropy (structured)   (b) Coverage rate (structured)

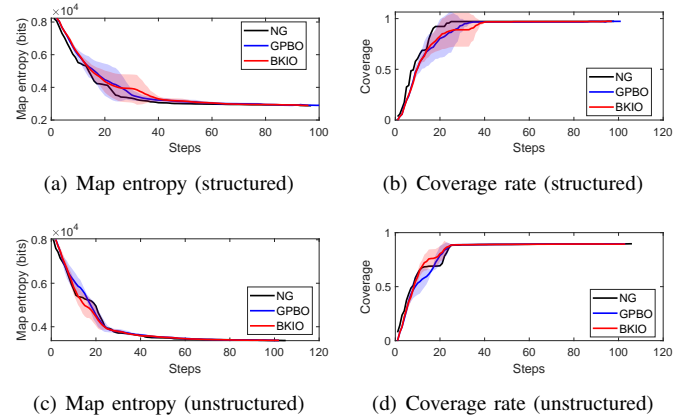(c) Map entropy (unstructured)   (d) Coverage rate (unstructured)

Fig. 3. Map entropy and coverage results of synthetic structured and unstructured maps. BKIO method can achieve similar exploration performance as GPBO, even better than GPBO in the unstructured map, and with less total exploration time.

Time efficiency analysis: As in Table I, we further analyze the time costs of both CRMI evaluation and exploration per
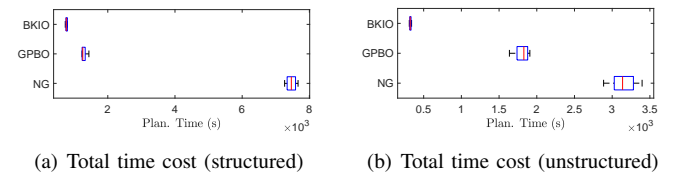

(a) Total time cost (structured)   (b) Total time cost (unstructured)

Fig. 4. Exploration time cost using different methods in synthetic maps.

TABLE I
TIME COST COMPARISON PER STEP OF DIFFERENT METHODS

Methods	Synthetic structured map	Synthetic unstructured map	Seattle map [34]
NG [6]	28.31±1.49 / 51.05±2.66	40.59±3.04 / 71.72±3.27	53.02±5.52 / 101.15±3.11
GPBO	7.25±0.98 / 10.27±1.07	9.70±1.31/ 17.08±2.24	11.97±1.10 / 21.65±1.94
BKIO	**2.71±0.36 / 5.14±0.48**	**2.98±0.43/ 9.95±1.17**	**3.22±0.42 / 10.54±1.09**

Note: Time cost (in seconds) of CRMI evaluation per step / Time cost of exploration per step.

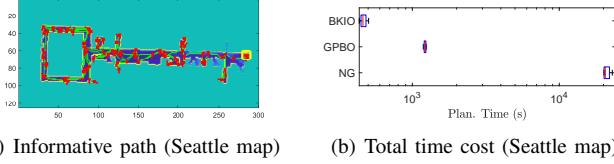(a) Informative path (Seattle map) (b) Total time cost (Seattle map)

Fig. 5. BKIO-based robot exploration in the Seattle map. *Yellow square*: start and end points. Note that the y-axis uses log10 representation.

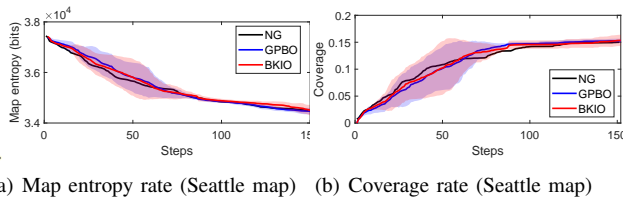(a) Map entropy rate (Seattle map) (b) Coverage rate (Seattle map)

Fig. 6. Map entropy and coverage results of the Seattle map. Note that the entropy and coverage would not be 0 bit and 100%, respectively, since plenty of thick walls and obstacles are also counted, not only the inside free areas, as in Fig. 5(a).

iteration in each method. Both BKIO and GPBO are much more efficient than the NG method in each step, and both the time costs of CRMI evaluation and exploration get a significant reduction. In addition, BKIO shows higher time efficiency performance than GPBO in all scenes.

For example, compared to the NG method, BKIO decreases the information gain evaluation time by about 90.4% in the structured map, compared with the one of 74.4% using GPBO. These performance differences still prevail in unstructured and large cluttered maps, e.g., BKIO decreases the evaluation time costs per step by about 92.6% and 93.9% in the more complex maps of unstructured and Seattle, much better than 76.1% and 77.4% of GPBO, respectively. The box plots depicting the entire exploration time cost in Fig. 4 and Fig. 5(b) also support the above efficiency analysis.

Experimental study of different N_{epoch} values: Here we will investigate the influence of N_{epoch} on time efficiency and quality, presenting the experimental study results in synthetic and dataset simulations, including some results about time efficiency using different N_{epoch}.

To study this, we use three representative N_{epoch} values of 1, 30, and 60 for BKIO, where $N_{\text{epoch}} = 1$ means the BO is only conducted once (i.e., only use BKI itself, denoted as 'BKIO-1'), $N_{\text{epoch}} = 30$ is what we use in this paper ('BKIO-30'), and $N_{\text{epoch}} = 60$ means the BO is conducted excessively ('BKIO-60').

We present the results of time efficiency in the following Table II and the exploration performance (map entropy reduction and coverage rate) in the following figures (Fig. 7). The results show that though the time efficiency of BKIO-1 significantly outperforms BKIO-30 and BKIO-60, the resulting exploration

performance of BKIO-1 is the worst of all in the following Fig. 7, ending the exploration too early.

BKIO-60 shows the best exploration performance but spends much more time than BKIO-30, while BKIO-30 performs similarly to BKIO-60, achieving a better balance of time efficiency and exploration performance. This phenomenon also implies that the performance improvement of higher N_{epoch} value has a marginal effect.

To balance the time efficiency and exploration performance, we choose the $N_{\text{epoch}} = 30$ for BKIO in the following simulations and real-world experiments.

We have also studied how N_{epoch} influences GPBO using the same value settings. Similarly, the results in Fig. 8 and Table III still show that $N_{\text{epoch}} = 30$ is a better choice in practice.
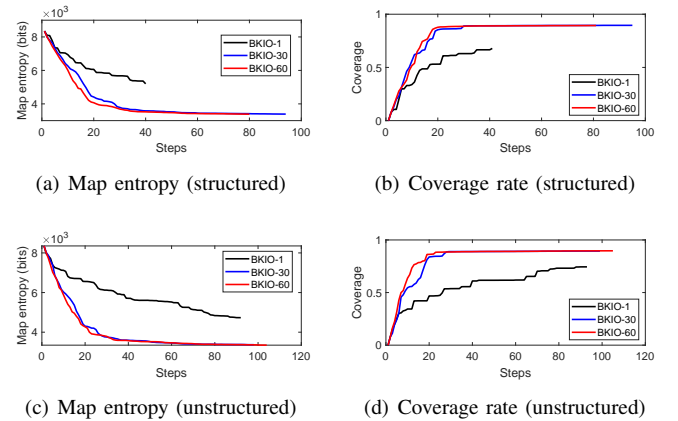

(a) Map entropy (structured) (b) Coverage rate (structured)

(c) Map entropy (unstructured) (d) Coverage rate (unstructured)

Fig. 7. BKIO exploration performance using different N_{epoch} values.

TABLE II
BKIO TIME COST (IN SEC) USING DIFFERENT N_{epoch} VALUES.

Methods	Structured map	Unstructured map
BKIO-1	0.08 / 2.47	0.14 / 2.75
BKIO-30	**2.27 / 5.05**	**3.07 / 9.61**
BKIO-60	4.62 / 8.85	7.96/ 12.19

Note: CRMI evaluation/exploration costs per step.

TABLE III
GPBO TIME COST (IN SEC) USING DIFFERENT N_{epoch} VALUES.

Methods	Structured map	Unstructured map
GPBO-1	0.23 / 3.84	0.44 / 4.82
GPBO-30	**6.57 / 11.36**	**7.65 / 15.96**
GPBO-60	9.36 / 16.33	9.31/ 20.04

Note: CRMI evaluation/exploration costs per step.

Local trade-off analysis: Here we use an intuitive example in the structured scene to show how the UCB of Eq. (7) improves the trade-off between exploration and exploitation locally at each exploration step.
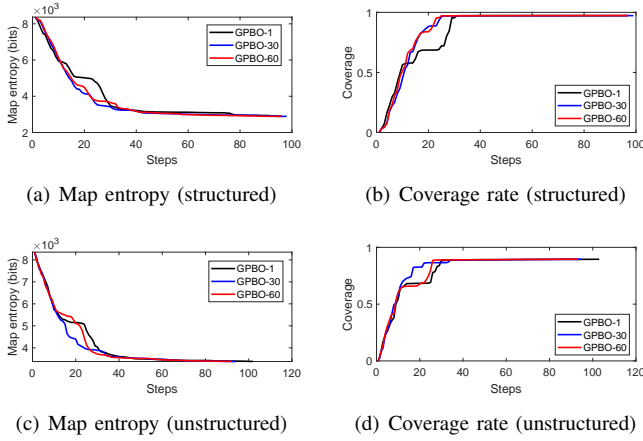
(a) Map entropy (structured)   (b) Coverage rate (structured)

(c) Map entropy (unstructured)   (d) Coverage rate (unstructured)

Fig. 8. GPBO exploration performance using different N_{epoch} values.
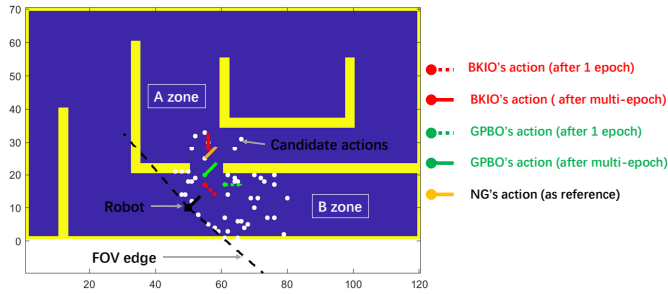

Fig. 9. Illustration of the trade-off between exploration and exploitation.

As in Fig. 9, the robot stands (black square with a black bar) at a fork road and decides where to go next. The NG method evaluates all candidate actions and suggests the best action (orange bar, max CRMI: 76.95 bits) moving towards the A zone. In contrast, the Bayesian optimization-based GPBO and BKIO methods suggest different actions before and after multiple epoch optimization. GPBO and BKIO initially suggested the action towards the B zone, marked by the green and red dashed lines, respectively (CRMI values: 65.12 and 67.57). After multiple epoch optimization, the results of GPBO and BKIO (green and red bars, respectively) have been improved (CRMI values: 71.69 and 73.44) and got closer to NG's suggested action. This means that, given limited explicitly evaluated samples, BO-based methods can avoid being stuck in the local optimality of max CRMI and explores actions with high prediction uncertainty, i.e., the multi-epoch BO chooses the promising actions suggested by the UCB function (Eq. (7)) and evaluates them explicitly, then adds them to the training set and improves the surrogate predictive model between actions and their CRMI.

Finally, using several epochs, the predicted results get closer to the ground truth, i.e., the local trade-off between exploration and exploitation in CRMI inference at each step can be achieved using the UCB function.

B. Real-world Experiments

We deploy our proposed BKIO and GPBO on the Fetch [2] mobile robot running C++/Python code in the Robot Operating System (ROS). As shown in Fig. 1, we conduct experiments in an approximately $25 \mathrm{~m} \times 35 \mathrm{~m}$ real-world office environment in Level 11, Building 2 of the University of Technology Sydney. There are many chairs, tables, and glass walls in the office and kitchen zones, which would hinder the ego perception and motion of the robot. The FOV of the LiDAR is 220 degrees, and the max sensing range is 25 m. Note that the robot explores the environment autonomously and only uses the laser scan data from the SICK 2D LiDAR (15 Hz) without cameras. The max sensing range is set to 8 m, forcing this 'sensing-limited' robot to explore longer.

Here we use the Cartographer [35] as the localization module and CRM [8] as the mapping module in the exploration framework. The robot calls for a self-defined ROS service to evaluate the CRMI of each action generated at the extracted frontiers, then decides where to go at the next step until the exploration ends when the travel cost reaches 500 m or the expected CRMI of the best action is lower than the predefined threshold of 10 bits. An example video is available here [3].

Fig. 11 shows the comparative evolution results of map entropy and explored areas using three different methods as the travel distance and total exploration time increase. The map entropy and coverage curves in Fig. 11(a) and (b) show these methods all complete the exploration finally. In Fig 11(a), the map entropy curves of the three methods w.r.t the traveled distance are similar, but the explored areas vary a lot in Fig. 11(b). This is mainly because we determine a map cell as already explored if its expected occupancy deviates from 0.5, otherwise unexplored (bool 0 or 1) when computing the covered area; but when computing the map entropy, we use the continuous expected occupancy (in the interval [0,1]). The rapid increase of explored areas in Fig. 11(b) and Fig. 11(d) also implies our BKIO can drive the robot to explore more unknown areas rapidly, given a limited distance budget, and get an environmental sketch for the subsequent robot application.

Fig. 11(b) shows that the GPBO travels a longer distance at the start stage but gets the less explored area than others since it moves in the corridors for a long distance and explores more aggressively and roughly in the later stage. In Fig. 11(d), we can observe that the BKIO and GPBO methods outperform the NG in terms of exploration time cost significantly (79.9% and 38.6% reduction, respectively), where BKIO performs better in explored areas than the others. Fig. 11(c) and (d) also show our methods conduct much more efficient exploration in map uncertainty reduction and exploring unknown areas.

Moreover, as in Fig. 11(d), BKIO reduces about 67.3% total time cost to end the exploration and increases about 21% covered areas than GPBO. Further, the mean time costs of each exploration step are 12.2095±0.6383s, 0.2007±0.1362s, and 0.0038±0.0033s for NG, GPBO, and BKIO, respectively. This also evidences the advantage of our proposed methods.

The exploration sequence differences of the three methods are shown in Figure 10, where the overall sequences of NG and

---

[2] http://docs.fetchrobotics.com
[3] https://youtu.be/sOW4fuaAwT8

GPBO in this figure are A-B-G-C-D-F-E-A and A-B-A-E-F-D-C-G-B-A, respectively. Both BKIO and NG began with A-B-G-C, while GPBO began with A-B-A-E. There are mainly two reasons accounting for this phenomenon. Firstly, the best actions suggested by different methods at each exploration step vary because the information evaluation accuracy differs, such as GPBO vs. BKIO (w.r.t NG as the ground truth). This difference accumulates over the exploration steps (e.g., choosing different intersections) and affects the resulting exploration sequence. These different choices made by the three methods will significantly change the exploration sequence. This phenomenon also exists in the simulation; referring to the local trade-off analysis in Section IV-A, Fig. 9 shows that all methods choose different actions at the intersection since the predicted CRMI values are unequal.

Though, the overall exploration metrics such as map entropy, explored area, total time cost, travel distance, and average exploration time cost can still help us assess the performance of different methods. Secondly, for each method,
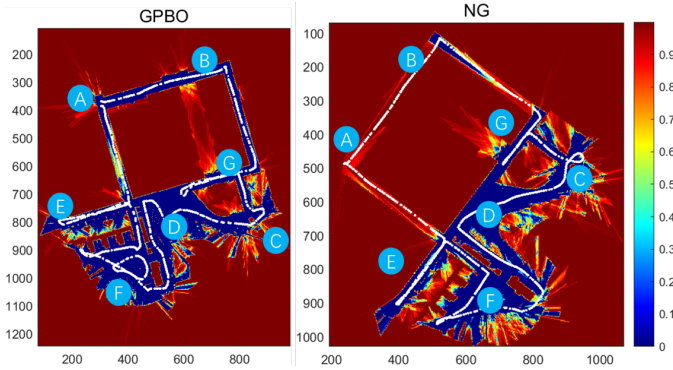

Fig. 10. GPBO and NG exploration results.

the CRMI-based objective function values of several different actions will be equal in rare cases, such as multiple actions all facing the open unknown space or parallel to the wall. The best action can only be selected from these actions randomly; we also try to improve this in our next work, such as using adaptive sampling or improving frontier extraction.

C. Discussion

As presented in previous experiments of typical scenes, our proposed methods (GPBO and BKIO) have shown desired exploration performance in efficiency and accuracy, especially the more efficient BKIO. Moreover, we use the public packages (e.g., scikit-learn and NumPy) to implement GP, but the BKI code is non-optimized. Hence we believe our methods' above performance improvement trend is not biased and would be more salient when facing more complex scenes.

The significant time-saving benefits for online exploration tasks greatly, especially when using computationally limited or energy-limited platforms. In other words, we offer two options for efficient information-based robot exploration tasks in unstructured and relatively large areas.

Another thing we should clarify is that we are not aiming to propose a new exploration framework. The exploration framework we use in this paper (i.e. Algorithm 1) stems from depth-first search, it is just used to run BKIO, GPBO, and

(a) Map entropy (w.r.t distance) (b) Covered area (w.r.t distance)

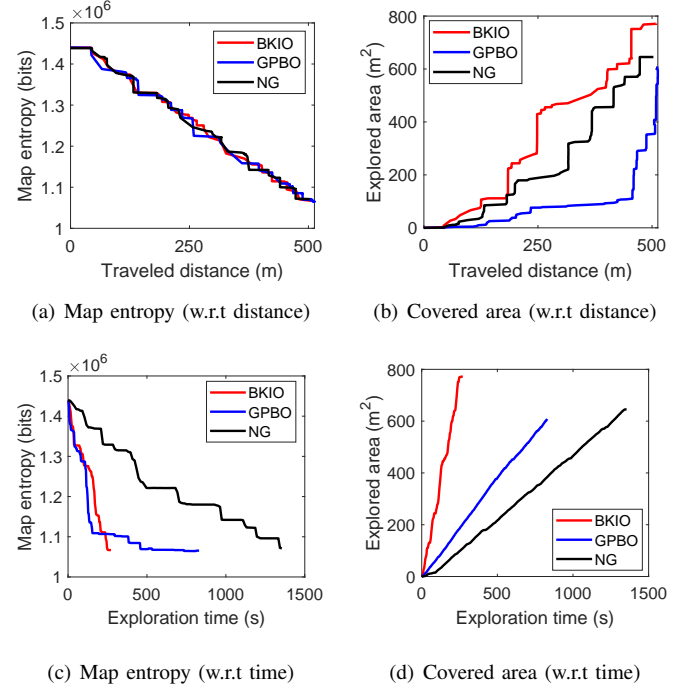(c) Map entropy (w.r.t time) (d) Covered area (w.r.t time)

Fig. 11. Real-world map entropy and coverage results w.r.t robot travel distance and total time cost, respectively.

NG methods. In other words, we can use another exploration framework instead, such as the sampling-based incrementally-informative graph [10]. We have also tested our methods in both sampling-based (Section IV-A) and frontier-based (Section IV-B) exploration. Hence we would not discuss the advantages and disadvantages of these frameworks because we focus on decreasing the evaluation time cost at each exploration step and the time cost of the whole exploration.

Moreover, the information metric is also not limited to CRMI. One can apply our methods to other information metrics for robot exploration, such as CSQMI [11] and GP-based MI (GPMI) [10].

In some experiments (e.g., Fig. 3), one can also find NG sometimes performs worse during the exploration (14-21 steps). The underlying reason is mainly that NG completely pursues the action maximizing the expected CRMI (i.e., *exploitation*), but may neglect more promising actions that have less expected MI currently (i.e., *exploration*). This myopic behavior may lead to undesired paths or even worse. Our future work will also study the non-myopic informative path planning in our next work to realize the global trade-off between exploration and exploitation.

V. CONCLUSIONS

This paper mainly contributed to new learning-based approaches with high efficiency for information-theoretic robot exploration in unknown environments. In particular, an information gain evaluation method for inferring the CRMI of numerous sampled robot actions is proposed based on GP and BO. With the Bayesian kernel inference-based BO, the time complexity of CRMI prediction decreases to a logarithm level. An informative objective function integrating the predicted

CRMI and uncertainty is also used to ensure the bounded regret and local trade-off between exploration and exploitation at each step. The proposed method also gets verified under an autonomous exploration framework by extensive simulations and real-world experiments in different scenes, which reveals the proposed BKIO and GPBO exploration methods outperform the non-learning ones overall in efficiency without losing much exploration performance, especially in unstructured and large cluttered scenes. This would also benefit other information-based exploration methods stuck by efficiency issues. Future work mainly involves the extension to 3D exploration and non-myopic planning.

REFERENCES

[1] H. Azpúrua, M. F. M. Campos, and D. G. Macharet, "Three-dimensional terrain aware autonomous exploration for subterranean and confined spaces," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 2443–2449.

[2] P. Stankiewicz, Y. T. Tan, and M. Kobilarov, "Adaptive sampling with an autonomous underwater vehicle in static marine environments," *Journal of Field Robotics*, vol. 38, no. 4, pp. 572–597, 2021.

[3] B. J. Julian, S. Karaman, and D. Rus, "On mutual information-based control of range sensing robots for mapping applications," *The International Journal of Robotics Research*, vol. 33, no. 10, pp. 1375–1392, 2014.

[4] B. Charrow, S. Liu, V. Kumar, and N. Michael, "Information-theoretic mapping using cauchy-schwarz quadratic mutual information," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 4791–4798.

[5] Z. Zhang, T. Henderson, S. Karaman, and V. Sze, "FSMI: Fast computation of Shannon mutual information for information-theoretic mapping," *The International Journal of Robotics Research*, vol. 39, no. 9, pp. 1155–1177, 2020.

[6] Y. Xu, R. Zheng, M. Liu, and S. Zhang, "CRMI: Confidence-rich mutual information for information-theoretic mapping," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6434–6441, 2021.

[7] Y. Xu, R. Zheng, S. Zhang, and M. Liu, "Confidence-rich localization and mapping based on particle filter for robotic exploration," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 4471–4477.

[8] A.-A. Agha-Mohammadi, E. Heiden, K. Hausman, and G. Sukhatme, "Confidence-rich grid mapping," *The International Journal of Robotics Research*, vol. 38, no. 12-13, pp. 1352–1374, 2019.

[9] G. A. Hollinger and G. S. Sukhatme, "Sampling-based robotic information gathering algorithms," *The International Journal of Robotics Research*, vol. 33, no. 9, pp. 1271–1287, 2014.

[10] M. G. Jadidi, J. V. Miró, and G. Dissanayake, "Sampling-based incremental information gathering with applications to robotic exploration and environmental monitoring," *The International Journal of Robotics Research*, vol. 38, no. 6, pp. 658–685, 2019.

[11] B. Charrow, V. Kumar, and N. Michael, "Approximate representations for multi-robot control policies that maximize mutual information," *Autonomous Robots*, vol. 37, no. 4, pp. 383–400, 2014.

[12] B. Charrow, G. Kahn, S. Patil, S. Liu, K. Goldberg, P. Abbeel, N. Michael, and V. Kumar, "Information-theoretic planning with trajectory optimization for dense 3D mapping." in *Robotics: Science and Systems*, vol. 11, 2015, pp. 3–12.

[13] S. Bai, F. Chen, and B. Englot, "Toward autonomous mapping and exploration for mobile robots through deep supervised learning," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 2379–2384.

[14] F. Chen, J. D. Martin, Y. Huang, J. Wang, and B. Englot, "Autonomous exploration under uncertainty via deep reinforcement learning on graphs," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 6140–6147.

[15] T. Wang, R. Liao, J. Ba, and S. Fidler, "Nervenet: Learning structured policy with graph neural networks," in *2018 International Conference on Learning Representations (ICLR)*, 2018.

[16] R. Marchant and F. Ramos, "Bayesian optimisation for informative continuous path planning," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 6136–6143.

[17] R. Oliveira, L. Ott, V. Guizilini, and F. Ramos, "Bayesian optimisation for safe navigation under localisation uncertainty," in *International Symposium of Robotics Research*. Springer, 2020, pp. 489–504.

[18] G. Francis, L. Ott, R. Marchant, and F. Ramos, "Occupancy map building through Bayesian exploration," *The International Journal of Robotics Research*, vol. 38, no. 7, pp. 769–792, 2019.

[19] S. Bai, J. Wang, K. Doherty, and B. Englot, "Inference-enabled information-theoretic exploration of continuous action spaces," in *International Symposium of Robotics Research*. Springer, 2015, pp. 419–433.

[20] S. Bai, J. Wang, F. Chen, and B. Englot, "Information-theoretic exploration with Bayesian optimization," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 1816–1822.

[21] W. R. Vega-Brown, M. Doniec, and N. G. Roy, "Nonparametric Bayesian inference on multivariate exponential families," *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[22] V. Peretroukhin, W. Vega-Brown, N. Roy, and J. Kelly, "PROBE-GK: Predictive robust estimation using generalized kernels," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 817–824.

[23] C. Richter, W. Vega-Brown, and N. Roy, "Bayesian learning for safe high-speed navigation in unknown environments," in *International Symposium on Robotics Research*. Springer, 2015, pp. 325–341.

[24] T. Shan, J. Wang, B. Englot, and K. Doherty, "Bayesian generalized kernel inference for terrain traversability mapping," in *Conference on Robot Learning*. PMLR, 2018, pp. 829–838.

[25] K. Doherty, T. Shan, J. Wang, and B. Englot, "Learning-aided 3-D occupancy mapping with Bayesian generalized kernel inference," *IEEE Transactions on Robotics*, vol. 35, no. 4, pp. 953–966, 2019.

[26] L. Gan, R. Zhang, J. W. Grizzle, R. M. Eustice, and M. Ghaffari, "Bayesian spatial kernel smoothing for scalable dense semantic mapping," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 790–797, 2020.

[27] ——, "Bayesian spatial kernel smoothing for scalable dense semantic mapping," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 790–797, 2020.

[28] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2005.

[29] S. T. O'Callaghan and F. T. Ramos, "Gaussian process occupancy maps," *The International Journal of Robotics Research*, vol. 31, no. 1, pp. 42–62, 2012.

[30] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," in *Proceedings of the 27th International Conference on Machine Learning*. OMNI Press, 2010.

[31] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, no. 2, pp. 235–256, 2002.

[32] L. Gan, Y. Kim, J. W. Grizzle, J. M. Walls, A. Kim, R. M. Eustice, and M. Ghaffari, "Multitask learning for scalable and dense multilayer bayesian map inference," *IEEE Transactions on Robotics*, vol. 39, no. 1, pp. 699–717, 2023.

[33] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with Rao-blackwellized particle filters," *IEEE Transactions on Robotics*, vol. 23, no. 1, pp. 34–46, 2007.

[34] A. Howard and N. Roy, "The robotics data set repository (radish)," 2003. [Online]. Available: http://radish.sourceforge.net/

[35] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2d LIDAR SLAM," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 1271–1278.