Learning-based Methods for Adaptive Informative Path Planning

Marija Popović[a,b,*], Joshua Ott[c], Julius Rückin[b], Mykel J. Kochenderfer[c]

[a]*MAVLab, Faculty of Aerospace Engineering, TU Delft, Kluyverweg 1, Delft, 2629 HS, The Netherlands*
[b]*Cluster of Excellence PhenoRob, Institute of Geodesy and Geoinformation, University of Bonn, Niebuhrstr. 1A, Bonn, 53113, Germany*
[c]*Stanford Intelligent Systems Laboratory, Department of Aeronautics and Astronautics, Stanford University, 496 Lomita Mall, Stanford, 94305, California, United States of America*

Abstract

Adaptive informative path planning (AIPP) is important to many robotics applications, enabling mobile robots to efficiently collect useful data about initially unknown environments. In addition, learning-based methods are increasingly used in robotics to enhance adaptability, versatility, and robustness across diverse and complex tasks. Our survey explores research on applying robotic learning to AIPP, bridging the gap between these two research fields. We begin by providing a unified mathematical problem definition for general AIPP problems. Next, we establish two complementary taxonomies of current work from the perspectives of (i) learning algorithms and (ii) robotic applications. We explore synergies, recent trends, and highlight the benefits of learning-based methods in AIPP frameworks. Finally, we discuss key challenges and promising future directions to enable more generally applicable and robust robotic data-gathering systems through learning. We provide a comprehensive catalog of papers reviewed in our survey, including publicly available repositories, to facilitate future studies in the field.

Keywords:
Informative path planning, Robot learning, Active learning

1. Introduction

The field of robotics has witnessed remarkable advancements in recent years, fueled by the growing need for automation and the increasing complexity of tasks required in various application domains. One of the key challenges in robotics is AIPP, which involves planning a trajectory for an autonomous robot to follow that maximizes the information acquired about an unknown environment while simultaneously respecting its resource constraints. This problem is of critical importance in applications such as environmental monitoring, exploration, search and rescue, and inspection across ground, aerial, and aquatic domains [1–7].

Despite its importance, AIPP is a challenging problem due to the inherent complexity of modeling and predicting new information in the environment, as well as the need to balance the exploration of unknown areas with the exploitation of newly acquired data [8]. Additionally, noisy sensor measurements introduce uncertainties into the data acquisition process, while actuation uncertainty can lead to deviations from planned trajectories. Moreover, real-world environments are often highly dynamic, which further complicates the modeling process and makes prediction challenging. Decisions must be made in a sequential manner to allow for continual refinement as more information becomes available during a mission.

Conventional approaches such as static pre-computed paths tend to fail on AIPP problems because they often rely on strong assumptions about the environment and cannot adapt to uncertainties or changes online [9]. Additionally, these approaches scale poorly to large, complex environments, and may not effectively account for the constraints and capabilities of the robot itself. These conventional solutions applied to the AIPP problem have been limited by computational complexity, lack of adaptability, and an inability to generalize across diverse environments and problems [10]. The growing popularity of learning-based methods has led to a renewed focus on applying these new techniques to AIPP, offering the promise of more flexible, adaptive, and scalable solutions.

In this survey paper, we present a new perspective on AIPP by exploring the potential of emerging robot learning techniques in addressing the challenges of AIPP. Different learning-based methods allow for constructing path planning algorithms that can naturally account for the inherent uncertainties and dynamic changes within a given environment. This adaptability facilitates more robust and efficient operations in complex environments because the robot can optimize its path based on its evolving understanding of its surroundings. This not only improves task performance, but also enhances resilience and versatility since the robot can react more effectively to new information about its environment.

*Corresponding author.
Email addresses: m.popovic@tudelft.nl (Marija Popović), jott2@stanford.edu (Joshua Ott), jrueckin@uni-bonn.de (Julius Rückin), mykel@stanford.edu (Mykel J. Kochenderfer)

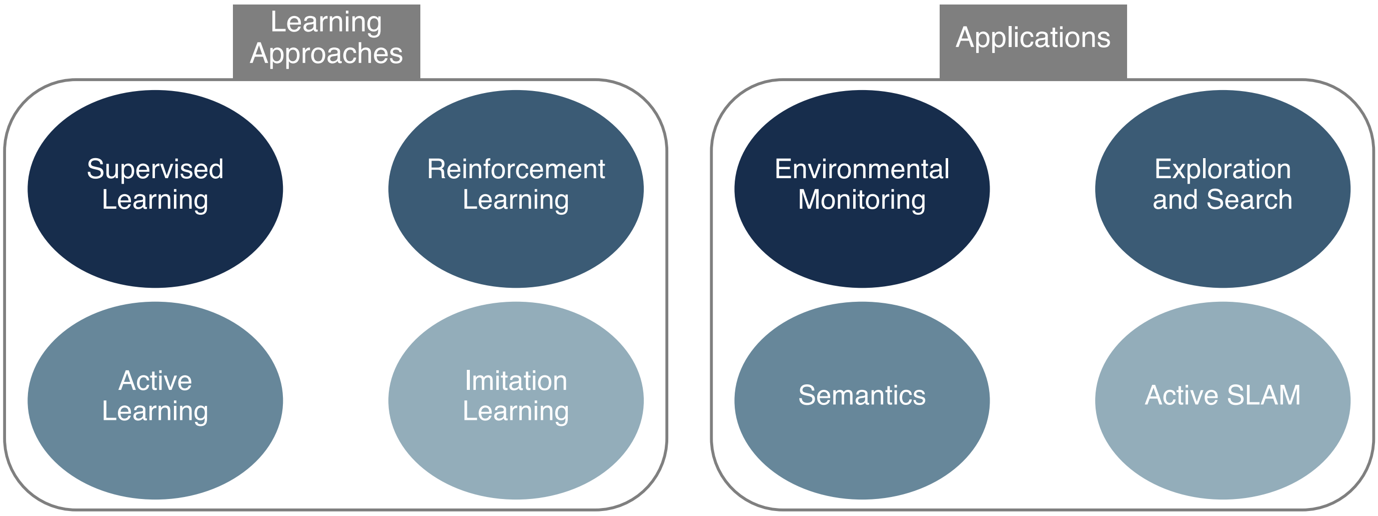

Fig 1: Overview of our survey paper. We review related work in learning-based methods for adaptive informative path planning (AIPP) from the perspectives of learning-based approaches (left) and practical applications (right).

Fig. 1 shows an overview of the main topics addressed in our survey paper. Our review includes different aspects of learning: supervised learning, reinforcement learning, imitation learning, and active learning. We focus on how these techniques can be used for AIPP in robotics. Furthermore, we discuss relevant application domains, such as environmental monitoring, exploration and search, semantic scene understanding, and active simultaneous localization and mapping (SLAM).

While there have been previous surveys addressing aspects of AIPP [11–16], our work distinguishes itself by specifically focusing on the intersection of learning-based methods and AIPP. Previous literature reviews have focused on specific subfields of robotic learning. For instance, Taylor et al. [15] focus on active learning in robotics, highlighting methods suitable for the demands of embodied learning systems. Similarly, Argall et al. [17] address imitation learning, where robots develop policies from example state-to-action mappings, while Garaffa et al. [18] investigate reinforcement learning techniques that devise unknown environment exploration strategies for single and multi-robot exploration. Lauri et al. [19] exclusively examine the application of Partially Observable Markov Decision Processes (POMDPs) in robotics, and Mukherjee et al. [20] analyze robotic learning strategies for human-robot collaboration in industrial settings. Other surveys, such as those by Sung et al. [14], Aniceto and Vivaldini [16], and Bai et al. [11] explore different different applications of AIPP, such as environmental monitoring tasks or general path planning techniques [9–12, 21]. However, these works do not underscore the applicability of learning in adaptive scenarios.

The main objectives of this survey paper are to:

1. provide a comprehensive understanding and taxonomy of the current state-of-the-art in learning-based methods for AIPP as well as their applications;
2. introduce a standardized mathematical problem definition for AIPP tasks, which provides a unified foundation for understanding and comparing various learning-based methods and application domains;
3. identify potential avenues for future research by highlighting the limitations of current approaches.

We have cataloged the papers in our survey at `https://dmar-bonn.github.io/aipp-survey/`. In particular, we highlight papers that have open-source implementations available.

2. Mathematical Formulation

First, we develop an underlying mathematical formulation to encompass the works included in our survey. We study the general problem of adaptive informative path planning (AIPP). The goal is to find an optimal action sequence $\psi^* = (a_1, \dots, a_N)$ of N robot actions $a_i \in \mathcal{A}, i \in \{1, \dots, N\}$, where actions may be, e.g., acceleration commands, next robot poses, using a specific sensor, or returning to a charging station. The action sequence ψ^* maximizes an information-theoretic criterion $I(\cdot)$:

$$\psi^* = \underset{\psi \in \Psi}{\operatorname{argmax}}\ I(\psi), \text{ s.t. } C(\psi) \leq B\,, \tag{1}$$

where Ψ represents the set of all possible action sequences, the cost function $C : \Psi \to \mathbb{R}$ maps an action sequence to its associated execution cost, $B \in \mathbb{R}$ is the robot's budget limit, e.g., time or energy, and $I : \Psi \to \mathbb{R}$ is the information criterion, computed from the new sensor measurements obtained by executing the actions ψ.

Our problem setup considers a robot gathering sensor measurements in an initially or partially unknown environment. In this environment $\xi \subset \mathbb{R}^D$, AIPP requires online replanning during a mission to *adaptively* focus on areas of interest as they are discovered. Each point $\mathbf{x} \in \xi$, e.g., a D-dimensional pose within the environment, is characterized by its distinctive features, represented as $F(\mathbf{x}, t) \in \mathcal{F}$. Here, $\mathcal{F}$ is a feature space, which could include characteristics like the semantic class, temperature, or radiation level, and may vary over time. These features may fluctuate during the mission, as defined by the feature mapping function $F : \xi \times \mathbb{R} \to \mathcal{F}$. The notation used to describe our AIPP formulation is summarized in Table 1.

Areas of interest within the environment the robot operates in are characterized by their features, which indicate interesting phenomena as defined by the mission objectives. Examples of mission objectives include identifying victims in a post-disaster scenario or locating regions of high radiation levels in a nuclear facility monitoring task. This concept of adaptivity in an optimal action sequence ψ^* is reflected in the specific definition of the information-theoretic criterion I during the optimization problem presented in Eq. (1).

Importantly, the point features $F(\mathbf{x}, t)$, such as temperature, may change over time. Given that the initial environment is unknown, the robot's understanding of $F(\mathbf{x}, t)$ may need to be updated during the mission. This could result in updating beliefs regarding interesting regions over time as new sensor measurements are received. Thus, to ensure that the robot's behavior is adaptively informed, we may re-solve the constrained optimization problem in Eq. (1) in a computationally efficient manner during a mission.

3. Background

The generic structure of a system for AIPP is shown in Fig. 2. In an active sensing context, the framework processes raw data from a robot sensor to obtain measurements used to build a map of the robot's environment. During a mission, the AIPP planning algorithm uses the environment maps built online to optimize future action sequences for maximum gain in an information metric as defined by the utility function. This section describes methods for environment mapping, evaluation metrics, and standard benchmarks commonly used in AIPP frameworks.

3.1. Mapping

The AIPP problem assumes the environment ξ to be *a priori* (partially) unknown. Thus, the true feature mapping $F : \xi \times \mathbb{R} \to \mathcal{F}$ and associated true regions of interest are *a priori* (partially) unknown as well. This makes the original AIPP problem stated in Eq. (1) not only computationally challenging but also impossible to solve directly as exact knowledge about interesting regions requires ground truth access to the feature mapping F. Instead of solving Eq. (1) directly, we model a stochastic process $\tilde{F}$ over all possible feature mapping functions. At each mission time step t, the robot's belief about the true feature mapping F, i.e., the probability measure associated with $\tilde{F}$ conditioned on the action-observation history, is updated by:

$$p(F \mid z_1, \ldots, z_t, a_1, \ldots, a_t)\,, \tag{2}$$

where $\{z_1, \ldots, z_t\}$ and $\{a_1, \ldots, a_t\}$ are all previously collected measurements and executed actions, respectively. The belief about $\tilde{F}$ is used to approximate the true information criterion I in Eq. (1).

The robot's belief about $\tilde{F}$ in Eq. (2) may be updated online during a mission as new sensor data z_t arrives at time step t after executing a sensing action a_t. Note that the information criterion $I(\cdot)$ reflects the planning objective and is computed based on the map state. In general, the computation of certain information criteria, e.g. entropy, may change for different map representations, e.g. occupancy maps and Gaussian processes, but the key underlying theoretical concepts remain the same.

The measurements z_t obtained at time step t may be uncertain or noisy. They are assumed to be sampled from $p(z \mid F)$ representing a probabilistic sensor model. Furthermore, there may be uncertainty in localization and actuation, leading to imperfect estimation of the robot pose as measurements are taken. In Sec. 4 and Sec. 5, we discuss in detail how these different sources of uncertainty can be accounted for in the context of learning-based methods.

To compute the belief $\tilde{F}$ following Eq. (2) based on the collection of stochastic measurements $\{z_1, \ldots, z_t\}$, various methods for *environment mapping* are commonly used. Table 2 provides an overview of commonly used mapping methods and corresponding metrics derived from them used to evaluate the performance of AIPP methods. We consider four broad categories in our taxonomy: occupancy grid mapping [22], Gaussian processes [23], graph-based methods, and implicit neural representations [24]. In general, the chosen mapping method for a given AIPP scenario depends on the environment, task, and available computing resources. For instance, Gaussian processes are applicable for problems where the to-be-mapped features $F(\mathbf{x}, t)$ support continuous feature spaces $\mathcal{F}$ and the map supports continuous environment representations ξ, whereas occupancy grid mapping is suitable for discrete mapping tasks. In contrast, graph-based methods are typically used for active SLAM problems [25] where both environment mapping and robot localization are concurrent. Recently, implicit neural representations, such as NeRFs [24] and pixelNeRFs [26], are gaining popularity for AIPP without relying on an explicit environment map. Due to their lightweight nature, these methods enable conserving computational and memory resources during planning in 3D reconstruction tasks where a high level of detail is required.

3.2. Evaluation Metrics

The goal of AIPP is to adaptively acquire information about initially unknown environment features F by collecting uncertain or noisy measurements z_t during a mission,

Symbol	Meaning	Examples
$a_i \in \mathbb{A}$	robot action	next robot position or pose, sensing behavior
$\psi \in \Psi$	sequence of consecutive actions	—
$\mathbf{x} \in \xi$	areas/locations in an environment	surfaces, patches of terrain
$I : \Psi \to \mathbb{R}$	information-theoretic criterion	map entropy, mutual information
$F : \xi \times \mathbb{R} \to \mathcal{F}$	feature mapping function	semantic class, spatial occupancy, temperature, radiation level

Table 1: Notation associated with the adaptive informative path planning (AIPP) problem.

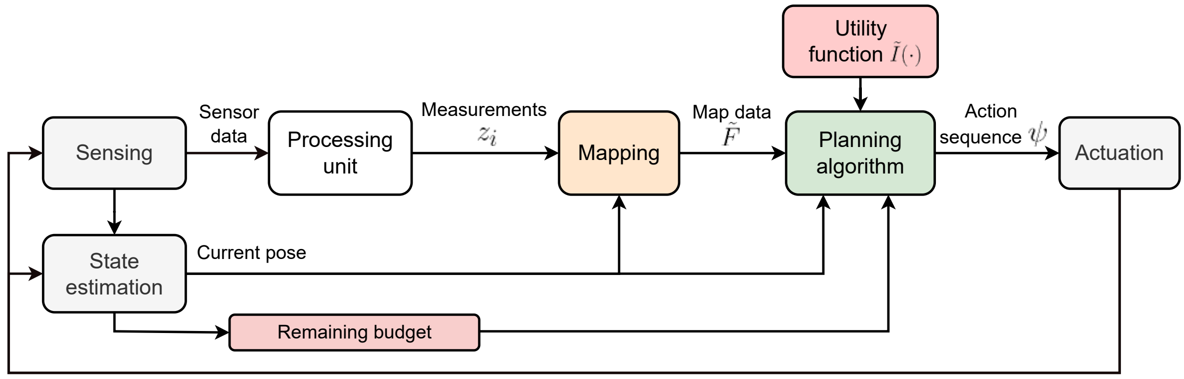

Fig 2: System diagram showing the key elements of a general AIPP framework. During a mission, a map $\tilde{F}$ of the robot's environment ξ is built using measurements z extracted from a sensor data stream. An AIPP algorithm leverages the map data to find the next action sequences ψ for the robot to maximize the information value $I(\psi)$, i.e. utility, of future measurements. The next action sequence ψ is executed by the robot, allowing for subsequent map updates in a closed-loop manner.

usually captured as a posterior map belief $\tilde{F}$ in Eq. (2). Because the robot mission goals are task- and application-dependent, e.g., searching for targets [62, 70, 74] or regions of high surface temperature [5, 27, 76], there is a lack of standard evaluation metrics used to quantify the performance of AIPP algorithms. Most works choose evaluation metrics closely related to their specified information criterion I in Eq. (1) because they reflect or approximate the mission goal. Table 2 lists commonly used evaluation metrics. Although these metrics vary significantly in the AIPP community, most works consider quantitative measures based on the map representation to compute the belief $\tilde{F}$, as reflected by our taxonomy.

Applications aiming to collect information about discrete features F, such as spatial occupancy [28, 29] or land cover classification [1, 3, 67], commonly use the entropy of the belief $\tilde{F}$ as a measure for the remaining uncertainty of the environment features [3, 27–31]. Lower entropy of $\tilde{F}$ indicates better AIPP performance. Following popular computer vision benchmarks [85, 86], some works evaluate the classification quality of $\tilde{F}$ by computing the mean Intersection-over-Union (mIoU) [36, 37], accuracy [36–38], or F1-score [27] of the maximum a posteriori estimate of the map belief $\tilde{F}$ given the ground truth features F.

Methods gathering information about continuous features F in an environment, e.g., bacteria level in a lake [65] or magnetic field strength [6], often use Gaussian process models [2, 6, 62, 65] or some variant of Bayesian filtering [3–5, 7] as the map representations. Although Gaussian processes allow for computing the differential entropy of $\tilde{F}$ in closed-form [87], it is common to use the covariance matrix trace to approximate the remaining uncertainty about the environment features and hence the information criterion I due to its computational efficiency. Despite relaxed computational requirements, most works evaluate AIPP performance based on the covariance matrix trace during offline evaluation as well [2, 3, 5, 61–64]. To evaluate the reconstruction quality of $\tilde{F}$, the root mean squared error (RMSE) between the maximum *a posteriori* estimate of $\tilde{F}$ and the ground truth F is usually reported [1–3, 5, 6, 61, 64, 67–69].

Evaluation methods for implicit neural representations differ in that they only rely on a trained neural network to represent the entire environment. The metrics can be grouped based on two aspects: the quality of the rendered images and the quality of the geometry of the reconstructed surface. Image quality metrics, e.g., peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM), and learned perceptual image patch similarity (LPIPS), are computed based on a set of evenly distributed test images in the environment [77–82]. To evaluate the reconstruction quality, the learned geometric

	Evaluation metrics	References
Occupancy grid	Entropy	[3, 27–31]
	Mutual information	[32, 33]
	Root mean squared error	[3]
	Mean absolute error	[34, 35]
	Mean intersection-over-union	[36, 37]
	F1-score	[27]
	Accuracy	[36–38]
	Precision/recall	[38]
	Coverage	[30, 34, 36–55]
	Observed surfaces	[56–58]
	Landmark error	[7]
Gaussian process	Entropy	[59]
	Expected improvement	[59, 60]
	Probability of improvement	[59]
	Covariance matrix trace	[2, 3, 5, 61–64]
	Mutual information	[65, 66]
	Root mean squared error	[1–3, 5, 6, 61, 64, 67–69]
	Mean log likelihood	[3]
	Distribution divergence	[70]
	Mean uncertainty	[68, 70]
	Upper confidence bound	[71]
Graph-based	Landmark uncertainty	[29, 34]
	Landmark error	[29]
	Localization uncertainty	[29, 61]
	Localization error	[61, 72]
	Target uncertainty	[8, 73, 74]
	Detection rate	[75]
	Path costs	[76]
Neural representation	Peak signal-to-noise ratio	[77–82]
	Structural similarity index measure	[77–80, 82]
	Learned perceptual image patch similarity	[78–80]
	Accuracy	[79, 80, 83]
	Completion	[79, 80, 83]
	Completion ratio	[79, 83]
	F1-score	[80, 84]
	Chamfer distance	[80]
	Precision	[84]
	Recall	[84]
	Surface coverage	[83, 84]

Table 2: Mapping methods and evaluation metrics used in AIPP.

representation is usually first post-processed, e.g., by using the marching cubes algorithm to extract a mesh. Then, a set of points is sampled on the mesh to compute various metrics, e.g., mesh accuracy or completion [79, 80, 84].

Some methods do not assume the robot to have perfect ground truth localization information during missions but additionally account for localization uncertainty while building the map belief $\tilde{F}$. These works propagate localization uncertainty into the map belief update in Eq. (2) using graph-based SLAM systems [29, 34, 61, 74] and Gaussian process variants supporting input uncertainty [61]. In addition to assessing the quality of the map, the quality of robot localization is also evaluated to determine whether the AIPP method can gather not only informative measurements to improve the environment map, but also measurements that improve localization to guarantee the construction of an accurate map. Some works compute the trajectory errors between the estimated and ground truth executed trajectory [34]. Other works track the localization uncertainty of the robot pose belief [29, 61], the localization error between the estimated and ground truth robot pose [61, 72], or the landmark errors between estimated and ground truth landmark poses [29].

An alternative problem formulation considers using AIPP to actively improve the sensor model reasoning about measurements z_t. In Eq. (1), this is done by linking the information criterion I to the value of new training data acquired during a mission. Such formulations generally do not aim to improve the belief of the map $\tilde{F}$ representing a physically quantifiable environmental phenomenon. Instead, works in this category target improving a neural network for semantic segmentation using RGB images [88–91] or object detection [92]. AIPP performance is evaluated using a held-out test set to quantify the neural network prediction performance [85, 86]. Commonly used metrics include mIoU [88–90], F1-score [90, 91], or average precision [92], depending on the prediction task.

Due to the sequential nature of the AIPP problem, most works consider a well-performing AIPP approach not only to show strong evaluation metrics after a mission is over but also to ensure fast improvement during a mission as the remaining mission budget B decreases. Trends in performance over mission time allow us to judge the efficiency of budget allocation, which is critical in AIPP as reflected by the constraints in Eq. (1). Common evaluation strategies include analyzing how the performance-based evaluation metrics evolve with the remaining mission budget, e.g., trajectory steps [2, 4, 6, 27, 62] or energy-based budgets [3, 5, 7, 68].

3.3. Benchmarks

Despite research efforts towards more generally applicable learning-based AIPP, most current methods are evaluated in application-dependent simulators and compared to task-dependent baseline algorithms. This results in a lack of standardized benchmark scenarios within the AIPP community. Benchmark scenarios are typically defined by two factors: (i) the choice of simulation environments and (ii) the choice of AIPP baseline algorithms against which the proposed methods are compared. Simulators fulfill a role analogous to datasets in the computer vision community as they define the problem setup in terms of the mission objective, evaluation environment, and robot configuration. AIPP baseline algorithms can be seen as paralleling widely established computer vision models, which are typically used to benchmark new models according to evaluation metrics.

3.3.1. Simulations

Simulators for AIPP evaluation make various interdependent design choices leading to highly specific simulation environments. First, AIPP methods are typically evaluated on a single fixed task, e.g., the RockSample task [7, 62], monitoring missions [2, 3, 5, 27, 68, 69], or reconstruction [12, 43, 48, 57, 79, 80, 83, 93]. Second, simulators differ in the robot configuration defined by the robot type, e.g., autonomous underwater vehicle (AUV) [73], unmanned aerial vehicle (UAV) [3], unmanned ground vehicle (UGV) [66], or a robot arm [43], as well as the sensors, e.g., depth sensors [40, 41, 46, 56], thermal camera [5], point-based sensors [2, 63, 65, 71], or abstractions of sensor models interpreting raw sensor data [3, 4, 59, 92, 94]. Third, works vary in their assumptions about the structure of environments, e.g., 2D terrains [2, 3, 5, 7, 62, 65] or 3D volumes [1, 41, 43], and robot workspaces, e.g., obstacle-free 2D [2, 95], obstacle-free 3D [1, 3, 5, 27, 59], or obstacle-aware 3D [41, 43, 56] workspaces. Last, simulators vary according to the datasets they use, e.g., synthetic environments or randomly generated distributions [2, 3, 5, 27, 28, 39, 65, 96], more realistic simulators [43, 45, 90, 97], or real-world remote sensing datasets [1, 3, 27, 49, 65, 68, 90]. Sec. 4 and Sec. 5 provide a comprehensive overview of these aspects for each paper studied in our survey. In general, diverse non-unified simulator choices result in task- and application-dependent evaluation protocols. As a result, it is difficult to compare the performance of different AIPP algorithms in a generalizable and fair fashion under consistent conditions.

3.3.2. Common and Classical Baselines

Despite the lack of standardized AIPP algorithms used for benchmarking new methods, some simple baseline algorithms have been established and used across different tasks and applications. These algorithms aim to foster exploration of the environment and use intuitive hand-crafted heuristics to optimize the evaluation metrics. We categorize approaches for exploration into two groups:

1. Geometric approaches, which pre-compute paths to maximize coverage of the initially unknown environment for uniform data collection. Examples include lawnmower-like grid patterns [9, 27, 88] or spiral- or circle-like patterns [3, 43, 97].
2. Random walk-like methods, which sample paths at random often combined with heuristics to efficiently manage the mission budget [30, 46, 55, 89, 98].

In addition to these simple non-adaptive benchmarks, many works consider variants of well-established planning algorithms in the AIPP context. Classical planners used for evaluation include branch-and-bound techniques [99] and sampling-based methods, e.g., Monte Carlo Tree Search (MCTS) [7], rapidly exploring information gathering (RIG) trees [68], or rapidly exploring random trees (RRTs) [100]. Other benchmarks include geometric strategies, e.g., greedy frontier-based exploration [101], or optimization-based routines, e.g., Bayesian optimization [102] or the covariance matrix adaptation evolution strategy (CMA-ES) [103]. To enable adaptive replanning during a mission as new sensor measurements arrive, these algorithms are commonly implemented in a fixed- or receding-horizon fashion.

4. Learning Approaches

This section surveys the AIPP literature from a learning-based perspective. We provide a taxonomy classifying relevant works based on the underlying learning algorithm they use, discuss pertinent aspects in each category, and integrate them in our mathematical problem definition for AIPP.

4.1. Supervised Learning

Traditionally, the components of the general AIPP framework in Fig. 2 are realized using hand-crafted models or heuristics, e.g., forward sensor models [3, 5, 100, 104] or hand-crafted information criteria [68, 75, 99, 105]. This makes it difficult to incorporate prior knowledge and the expected distributions of geometry and information in complex scenarios, as well as to achieve adaptability in new environments. To address these drawbacks, various methods have been proposed to learn different elements of mapping and planning through supervised training from labeled data.

Given a dataset of N training examples of the form $\{(x_1, y_1), \dots, (x_N, y_N)\}$, a supervised learning algorithm seeks to learn a ground truth function $g : X \to Y$, where X and Y are the input and output spaces, respectively, and $(x_i, y_i) \in X \times Y$ are dataset pairs. Table 3 provides a taxonomy of supervised learning applications in AIPP. These methods find diverse usage in both mapping and planning.

Supervised learning has been used for modeling environmental variables through Gaussian processes [1–3, 5, 6, 59, 60, 63–71]. As described in Sec. 3.1, Gaussian processes are often used in Eq. (2) due to their probabilistic and non-parametric nature, making them suitable for uncertainty-based information gathering with spatially complex underlying feature mappings F. In this context, supervised learning is used to compute the ground truth function $g = F$ from a training dataset $(x_i, z_i) \in \xi \times \mathcal{F}$, with $X = \xi$ and $Y = \mathcal{F}$, obtained by taking stochastic measurements z_i in the environment. Note that we use x_i to denote general training inputs and $\mathbf{x}_i$ to denote the robot pose. In Gaussian process mapping, each x_i corresponds to the pose $\mathbf{x}_i \in \xi$ introduced in Sec. 2. Example applications include mapping distributions of temperature [5, 27, 61], light [2, 4], water properties [64, 65], and vegetation [3, 67]. The learning process in implicit neural representations can be interpreted similarly. These methods learn a function that maps 3D coordinates $X = \xi$ to radiance values $Y = \mathcal{F}$, which represent color and light intensity, from a dataset of posed RGB [77, 78, 80–82] or RGB-D [79, 83, 84, 97] images.

For robotic exploration, several works focus on learning map completion from a partially observed map state, represented by its geometry [36, 44, 51, 52, 54, 57], topology [96], or semantics [38]. At mission time step t, these methods predict the evolution of a probabilistic map representation k steps into the future $\tilde{F}_{t+k}$. The dataset pairs

(x_i, y_i) used for learning map completion are application-dependent; for example, learning inputs can range from local observations [38] to the current global map state [106]. The key advantage of such approaches is their ability to extrapolate incomplete maps by minimizing a specific loss during training. This extrapolation enables better reasoning with limited information and occlusions compared to classical heuristic strategies that plan solely based on the map state at a given time step.

Several works have also investigate the benefits of using supervised learning in the planning algorithm. One straightforward approach is to learn the information criterion $I(\cdot)$ in Eq. (1) from a dataset of actions and their associated information gains observed in previous data gathering missions $(x_i, y_i) = (a_i, I(a_i))$, where $X = \mathcal{A}$ and $Y = \mathbb{R}$ [28, 56, 107]. To solve the AIPP problem in Eq. (1) directly, the learning process can depend on pairs that connect map states to corresponding actions. Each action a_i is created using a planning algorithm with full access to the ground truth data represented by g. Schmid et al. [39] and Zacchini et al. [45] propose learning a probabilistic model of informative views in the context of sampling-based planning. Alternatively, the next best viewpoint can be learned by reasoning about the shape of an object to be reconstructed [47, 48] or localization performance in the context of active SLAM [108]. When the dataset pairs (x_i, y_i) are generated through expert demonstrations, supervised learning can be thought of as a simple form of imitation learning (IL) as described in Sec. 4.3. By learning the information criterion $I(\cdot)$ from data, these approaches all bypass formulating it explicitly for online replanning.

A key requirement for all methods in Table 3 is the availability of reliable training data for model supervision. Some supervised learning methods in AIPP rely on open-source datasets, such as the indoor Matterport3D dataset [36, 51, 52] or outdoor 3D Street View dataset [56]. Works in environmental monitoring may use data from previously executed missions [1, 49, 65, 68], while others create custom synthetic environments approximating real-world scenarios [2, 3, 5, 6, 28, 39, 54, 60, 61, 63, 65, 71, 96]. However, the lack of realistic labeled training data for AIPP problems remains an open issue restricting the generalizability of existing methods to new environments and domains. Furthermore, since supervised learning models are trained only on static data, their applicability is limited in dynamic environments with changing obstacles or terrains or the target information distribution varies over time. Finally, as most deep learning models do not consider uncertainty in the learning predictions, they lack natural mechanisms to handle sensor noise or other sources of uncertainty. In Sec. 4.4, we discuss how active learning (AL) techniques can mitigate some of these issues.

4.2. Reinforcement Learning

Reinforcement learning (RL) is a branch of machine learning that equips an agent to learn from an environment by interacting with it and receiving feedback in terms of rewards or penalties. In AIPP, RL has emerged as an effective learning methodology, enabling a system to adapt its behavior based on an evolving understanding of its environment.

The problem of AIPP can be framed as a partially observable Markov decision process (POMDP) [112]. Formally, a POMDP is defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{O}, T, Z, R \rangle$, where:

- $\mathcal{S}$ is the *state* space;
- $\mathcal{A}$ is the *action* space;
- $\mathcal{O}$ is the *observation* space;
- $T(s' \mid s, a)$ is the *state transition function*;
- $Z(o \mid s', a)$ is the *observation function*;
- R is the immediate *reward function*.

In the context of AIPP, the true state of the environment is the actual physical attributes of the environment captured by $F(\mathbf{x})$ in Sec. 2, e.g., the locations of obstacles and areas of interest, while the observation z is the robot's current measurement of these attributes based on sensor readings.

The *state* space $\mathcal{S}$ is defined as the set of all possible feature mappings F in the environment and agent-specific information, e.g., its position or battery level. The *action* space $\mathcal{A}$ consists of all possible actions $a_i \in \mathcal{A}$ that the robot could take, e.g., moving to a given location, using a specific sensor, or returning to a charging station. The *observation* space $\mathcal{O}$ corresponds to the set of all possible sensor readings and derived interpretations the robot could have about the environment, e.g., semantic segmentation of RGB images.

The *state transition function* $T(s' \mid s, a)$ describes the probabilistic dynamics of the robot's state in the context of the AIPP problem. When the robot executes an action $a \in \mathcal{A}$ in its current state $s \in \mathcal{S}$, it moves to a new state $s' \in \mathcal{S}$. The inherent uncertainty of action outcomes and their impact on the environment is captured by the transition function. This function is used in the belief update, which refines the robot's understanding of the environment feature mapping F. Additionally, the *observation function* $Z(o \mid s', a)$ provides the likelihood of perceiving an observation $o \in \mathcal{O}$ subsequent to action a and arriving at state s'. This observation model, which is probabilistically linked to the feature mapping F through $z_t \sim p(z \mid F)$, is used to update the robot's belief state using Eq. (2).

The *reward function* $R(s, a)$ captures the trade-off between maximizing information gain, as described by the criterion $I(\cdot)$, and adhering to a budget constraint $C(\psi) \leq B$. While the objective function $I(\cdot)$ evaluates a sequence of actions, the *reward function* $R(s, a)$ evaluates individual actions. That is, the information criterion $I(\cdot)$ in AIPP accumulates the rewards $R(s, a)$ over a sequence of actions,

	Algorithm keyword	References
Architecture/method	Conditional variational autoencoder	[39]
	Convolutional neural network	[28, 36, 37, 39, 47–49, 52, 52, 54, 56, 58, 82, 96, 107, 109]
	Gaussian process	[1–3, 5, 6, 59, 60, 63–71, 110, 111]
	Implicit neural representation	[77–84, 97]
	Kernel density estimation	[45]
	Occupancy prediction network	[51]
	Scene completion network	[38]
	Transformer	[57, 108]
	Variational autoencoder	[44]
Training objective	3D semantic scene completion	[38]
	2D occupancy map completion	[36, 37, 44, 54]
	3D occupancy map completion	[51]
	Environment topology	[96]
	Environmental variable	[1–3, 5, 6, 59–61, 63–71, 109–111]
	Grasp affordance	[97]
	Informative view distribution	[39, 45]
	Localization score	[108]
	Minimum view subset for coverage	[58]
	Next-best view	[47, 48]
	Point cloud shape completion	[57]
	Radiance field	[77–81, 83, 84, 97]
	Required number of views	[82]
	Target point of interest	[49]
	Unexplored navigable area beyond map frontiers	[52]
	Utility/reward	[28, 39, 56, 107]
Mission objective	Coverage	[38, 39, 47, 48, 51, 52, 54, 96, 107]
	Estimate quantile values	[1]
	Find maximum field value	[59, 60]
	Maximize grasp quality	[97]
	Maximize localization accuracy	[108]
	Maximize probability of improvement/expected improvement	[59, 110, 111]
	Maximize observed surfaces	[57, 58]
	Minimize explored area to target point of interest	[49]
	Minimize map error	[6, 37, 69, 109]
	Minimize map uncertainty	[28, 36, 44, 45, 56, 59, 64, 66, 68, 110, 111]
	Minimize map uncertainty and robot localization uncertainty	[61]
	Minimize map uncertainty in high-interest areas	[2, 3, 5, 63, 65, 68, 71]
	Minimize map uncertainty in unclassified areas	[67]
	Minimize model uncertainty	[77–81, 83, 84]
	Minimize target localization uncertainty	[70]
	Object reconstruction based on PSNR	[82]
	Point-goal navigation	[36, 37]
Training data	3D Street View dataset	[56]
	Custom CAD models	[47, 48, 58, 79, 83]
	Custom synthetic dataset	[2, 3, 5, 6, 28, 39, 54, 60, 61, 63, 65, 71, 96]
	DTU dataset	[77]
	Heuristic parameter setting	[64, 70]
	Gibson dataset	[37]
	INRIA Aerial Image Labeling dataset	[107]
	KTH dataset	[44, 49]
	LLFF dataset	[78, 81]
	Matterport3D dataset	[36, 51, 52, 108]
	NeRF/Synthetic-NeRF dataset	[78–81, 84]
	NYU dataset	[38]
	Real-world environmental data	[1, 49, 59, 65–68]
	Realistic simulator	[45, 97]
	Regional Ocean Modeling System (ROMS) dataset	[109]
	ShapeNet dataset	[57, 77, 82]
	Tanks&Temples dataset	[80]

Table 3: Supervised learning methods for AIPP.

and can additionally incorporate a discount factor γ to reflect that future rewards are less valuable than immediate rewards:

$$I(\psi) = \sum_{t=1}^{N} \gamma^{t-1} R(s_t, a_t)\,. \tag{3}$$

In AIPP, the discount factor is typically set to $\gamma = 1$ for finite horizons with budget constraints.

The *policy* is a function $\pi : \mathcal{S} \to \mathcal{A}$. The concept of a policy and the general action sequence ψ in Eq. (1) are linked by $\psi = (\pi(s_1), \dots, \pi(s_N))$, where the states evolve based on the previous state and action according to $T(s' \mid s, a)$. The optimal policy can be queried to select an action sequence ψ^* maximizing the total expected discounted return, and hence the information gain, while being within the allowed budget.

Any POMDP can be viewed as an MDP that uses beliefs as states, also called a belief-state MDP or belief MDP. The *state* space of a belief MDP is the set of all beliefs $\mathcal{B}$. The *action* space is equivalent to that of the POMDP [112]. To connect this directly to Sec. 3, in AIPP, the robot's belief about the true state of the environment is denoted by $\tilde{F}$ in Sec. 3.1. This belief evolves as the robot gathers information about its environment, which starts as partially unknown and is gradually discovered through sensing and exploration.

4.2.1. Network Architectures and Training Algorithms

Various neural network architectures can be used in RL for AIPP, such as attention-based neural networks, convolutional neural networks (CNN), long short-term memory (LSTM) networks, and graph neural networks (GNN).

Attention-based neural networks, inspired by human visual attention, focus selectively on input data, which is useful for AIPP where the robot focuses on high-interest regions. CNNs excel at capturing spatial hierarchies in 2D occupancy grid maps. LSTMs, a type of recurrent neural network, learn long-term dependencies, which is ideal for temporally dependent problems in AIPP. GNNs operate on graph structures, making them most appropriate for graph-modeled environments.

Several RL algorithms can train these network architectures including proximal policy optimization (PPO), asynchronous actor-critic (A3C), soft actor-critic (SAC), advantage actor-critic (A2C), deep Q-network (DQN), double deep Q-network (DDQN), as well as many other variations and combinations, e.g. AlphaZero. These algorithms vary in exploration and exploitation, sample efficiency, computational needs, and stability.

Actor-critic methods use a value function estimate to guide optimization. The actor is the policy, while the critic is the value function. They train in parallel, differing in value function, advantage function, or action-value function approximation. Most methods focus on stochastic policies, but some support deterministic continuous actions. For example, PPO, A3C, SAC, and A2C follow an actor-critic approach with separate policy and value networks. In contrast, DQN and DDQN are critic-only value-based methods, where the optimal policy is inferred from the value function.

AlphaZero combines the actor-critic framework with Monte Carlo tree search (MCTS), using the network's policy to guide the search and the value function to evaluate leaf nodes. MCTS simulations generate a robust policy, considering a longer planning horizon [113, 114].

Various open-source implementations and benchmarks of these network architectures and training algorithms are widely available [115–119].

4.2.2. Reinforcement Learning Approaches

A variety of RL-based approaches have been explored in the literature. In Table 4, we distinguish the methods primarily by how they formulate each component of the POMDP architecture as well as how they learn policies.

All of the methods surveyed include some aspect of the current robot information in the state. The robot pose is often used [32, 41, 43, 59, 71], while other methods include additional robot information, e.g., energy level [7, 62], viewing direction [50], and operational status [120]. In addition to current robot information, all methods store some form of a spatial map in the state. The most popular spatial maps are Gaussian processes [2, 5, 6, 59, 62–64, 66, 69–71] and occupancy grids [27, 29, 30, 32–34, 40, 41, 46, 50, 53, 54, 121], both of which are described in Sec. 3.1. Another common component of the state is the previous position and sensor history. The history components can range from the executed trajectory [2] to the previous graph history [70] as well as the previous observations [4, 6]. Some methods implicitly capture the history of robot poses and measurements in the belief state action-observation history [7, 62] or by capturing the previously explored areas in the occupancy grid [27, 34, 53].

The *action* space determines the set of actions available to the robot in the current state. At the highest level, actions are distinguished by being discrete (e.g., move left or move right) or continuous (e.g., set velocity to 5 m/s). For example, Yang et al. [121] use a continuous action space consisting of longitudinal and lateral velocity commands while Lodel et al. [32] use a continuous 2D reference viewpoint for their local planner. Hüttenrauch et al. [72] use linear and angular velocity or acceleration commands. Continuous action spaces are not just restricted to movement commands. Bartolomei et al. [122] focus on semantic-aware active perception while reaching a goal target using an action space consisting of a set of weights to assign to each semantic class.

Continuous action spaces often significantly increase the computational complexity of POMDPs. In many cases, discrete action spaces are simpler to work with. When the spatial map is represented as an abstract graph, a common choice is for movement commands to be represented as movement from one graph node to another via the connecting edge [2, 4–7, 27, 35, 41, 53, 54, 59, 62–64, 66, 70, 73, 75, 120]. Another common approach relies on frontiers, which represent the boundary between known and unknown space in the environment and can be seen as a special case of a graph node action. The use of frontiers as movement actions does not necessarily require the spatial map to be represented as an abstract graph [29, 34, 40, 46]. Other discrete actions include selecting from a discrete set of sensors to employ [7, 62] as well as motion primitive commands [60, 71] which can be seen as a discretization of continuous actions where the agent is given a set of potential trajectories to execute which correspond to actions.

The *reward function* is the most problem-dependent component of the formulation. One of the core challenges that arise in the POMDP framework and, consequently, in AIPP, is the exploration-exploitation trade-off. The agent must select actions that balance between exploring unknown areas of the environment to increase its knowledge (exploration) and exploiting its current knowledge to find areas of interest within the constraints of its available resources (exploitation). This trade-off is intrinsically linked to the AIPP objective of maximizing the information criterion while respecting the cost constraint. Hence, it is a critical factor driving the learning process of the RL agent. Essentially, all AIPP methods use some measure of informativeness as the objective as discussed in Sec. 3.2. Map uncertainty reduction in high-interest areas is a very common RL reward in the AIPP literature [4, 5, 27, 35, 59, 60, 62, 63, 71, 73]. Some approaches use only map uncertainty reduction to spread measurements across the environment as uniformly as possible and minimize the estimation error [2, 6, 40, 41, 62, 64, 69, 70]. Coverage is a common surrogate metric for map uncertainty reduction since it only relies on computing the area uncovered as the robot travels through the environment, which is often faster than computing the Gaussian process variance online [30, 43, 46, 50, 54, 75, 98, 120]. While coverage can perform well on the map uncertainty reduction objective, it cannot explicitly reason about high-interest areas. The mutual information between observations and landmarks is another common way to measure map uncertainty reduction [32, 33, 66, 121]. Some methods also explicitly incorporate distance-based metrics in the reward function [29, 32, 34, 41, 53]. For example, Chen et al. [29, 34] and Cao et al. [41] assign greater rewards for actions that achieve the same uncertainty reduction in less distance.

Optimally solving POMDPs is computationally challenging [112, 123]. However, practical solutions can be obtained through various RL algorithms. We categorize AIPP policies based on their deployment inference strategies, distinguishing between zero-shot inference and expected discounted m-step search policies. Zero-shot inference policies are trained offline, considering the sequence of actions. The policy is trained to account for the cascading impact of each action on subsequent decisions, taking into consideration the consequences of actions and future dependencies during the offline training process. Upon deployment, these policies take the current belief state—updated to reflect the cumulative knowledge of the environment as per Eq. (2)—and directly output the next best action without additional planning. Conversely, policies using an expected discounted m-step search are dynamic in the sense that they actively consider a sequence of potential future actions during deployment. Instead of reacting to the current belief state, this approach projects forward, evaluating a decision tree of potential action sequences up to m steps ahead and factoring in the anticipated updates to the belief state for each action. The policy then chooses an action based on the strategy that maximizes the total expected reward over this lookahead horizon.

Yang et al. [121] use an attention-based multi-layer perceptron (MLP) which is trained using proximal policy optimization (PPO) to produce a zero-shot inference action of continuous longitudinal and lateral velocity with the goal of maximizing the mutual information between observations and landmark states. Cao et al. [2] take a slightly different approach by using an attention-based graph neural network and long short-term memory (LSTM) network with PPO to select the next graph node. Rückin et al. [5] use AlphaZero by combining Monte Carlo tree search (MCTS) with a convolutional neural network (CNN) to learn information-rich actions in adaptive data gathering missions. Chen et al. [33] uses deep Q-network (DQN) with a CNN to select the next location to visit from a set of randomly sampled 2D positions with the goal of maximizing the mutual information between received observations and the occupancy grid map. Chen et al. [34] uses the advantage actor-critic (A2C) algorithm with a graph neural network to select the next best frontier to visit with the goal of maximizing landmark uncertainty reduction and minimizing travel distance. Cao et al. [41] uses the soft actor-critic (SAC) algorithm with an attention-based graph neural network to output the next graph node to visit with the goal of maximizing the number of observed frontiers while minimizing travel distance. Other methods mainly differ in the training algorithms and the network architectures used with the most popular being PPO, A2C, SAC, and CNN, DQN, LSTM, respectively [2, 29, 32–34, 40, 41, 46, 54, 70, 93, 121].

In summary, RL offers a powerful framework for AIPP. By learning to optimize actions for maximum information gain, RL-based systems can adaptively and effectively explore unknown environments. However, RL algorithms also face challenges such as data-intensive training, sensitivity to reward design, and computational demands, which can affect their efficiency and generalizability in real-world robotics scenarios. Existing RL-based AIPP literature addresses these issues through advanced neural network architectures, sophisticated training algorithms, careful reward and policy design, and robust map representations. Additionally, most of these approaches are tested on environments very similar to those used for training. A promising direction for future work is thus to develop methods that can generalize to new environments.

	Description	References
State	Current robot information	[2, 4–8, 27, 29, 30, 32–35, 40, 41, 43, 46, 50, 53, 54, 59, 62–64, 66, 69–73, 75, 93, 98, 120–122, 124–129]
	Observation history	[4, 6–8, 27, 32, 46, 70, 93, 127, 130]
	Robot history	[2, 4, 29, 34, 53, 70, 93]
	Spatial map	[2, 4–8, 27, 29, 30, 32–35, 40, 41, 43, 46, 50, 53, 54, 59, 62–64, 66, 69–73, 75, 93, 98, 120–122, 124–126, 128, 129]
Map	2D Gaussian process	[2, 5, 6, 59, 62–64, 66, 69–71]
	2D occupancy grid	[27, 29, 32, 33, 40, 46, 50, 53, 54, 121, 128, 129]
	3D occupancy grid	[30, 34, 41]
	3D semantic occupancy map	[43, 122]
	Bayesian network	[126]
	Factor graph	[4]
	Gaussian Markov random field	[73]
	Landmark locations and uncertainties	[29, 34, 122, 125]
	Location graph	[7, 8, 72, 75, 124, 128, 129]
Action	Continuous action	[30, 32, 59, 72, 121, 122, 125, 127]
	Discrete action	[33, 35, 43, 50, 62, 69, 71, 93, 98, 120, 124, 126, 128, 130]
	Frontier	[29, 34, 40, 46]
	Graph node	[2, 4–8, 27, 35, 41, 53, 54, 59, 62–64, 66, 70, 73, 75, 120, 129]
Policy	Expected discounted m-step reward	[4, 5, 7, 30, 59, 62, 71, 73, 75, 126–129]
	Zero-shot inference	[2, 6, 8, 27, 29, 32–35, 40, 41, 43, 46, 50, 53, 54, 59, 63, 64, 66, 69, 70, 72, 93, 98, 120–122, 124, 125, 130]
Reward	Belief state mode	[7]
	Coverage	[30, 43, 46, 50, 54, 75, 98, 120, 127–129]
	Maximize inter-agent distance	[72]
	Maximize mutual information of observations and landmarks	[32, 33, 66, 121]
	Minimize landmark localization uncertainty	[29, 34, 93, 122]
	Minimize map uncertainty	[2, 6, 40, 41, 62, 64, 69, 70]
	Minimize map uncertainty in high-interest areas	[4, 5, 8, 27, 35, 59, 62, 63, 71, 73, 126]
	Size of bounding box for target detection	[130]
	Travel distance	[29, 32, 34, 41, 53, 124]
	Travel time	[125]

Table 4: Reinforcement learning methods for AIPP.

4.3. Imitation Learning

Reinforcement learning (RL) approaches typically assume that either the *reward function* is known or that rewards are received while interacting with the environment. For some applications, it may be easier for an expert to demonstrate the desired behavior rather than specify a reward function. Imitation learning (IL) refers to the case where the desired behavior is learned from expert demonstrations.

A simple form of IL can be thought of as a supervised learning problem from Sec. 4.1 typically referred to as behavioral cloning [131]. The objective is to train a stochastic policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$ where the policy π_θ is parameterized by θ. The training objective is to maximize the likelihood of actions from a dataset $\mathcal{D}$ of expert state-action pairs:

$$\underset{\theta}{\text{maximize}} \prod_{(s,a)\in D} \pi_\theta(a \mid s). \tag{4}$$

This approach, while straightforward, is limited by the quality and quantity of the expert demonstrations available. Moreover, it often generalizes poorly to states not encountered in the training dataset, leading to suboptimal or even erroneous behaviors under novel circumstances. More sophisticated IL strategies have been developed to address these limitations. These include methods combining IL with RL to refine policies beyond the initial demonstrations, approaches that incorporate active learning (AL) to query experts for demonstrations in states where the learner is uncertain, and algorithms that utilize inverse RL to infer the underlying reward function implied by the expert's behavior.

It is important to note the similarities and differences between IL and RL. Both IL and RL seek to find an optimal policy for a given task. However, the key distinction lies in how they learn this policy. RL aims to learn the policy by maximizing the expected discounted return, as illustrated in Eq. (3). In contrast, IL, as depicted in Eq. (4), assumes access to samples from an expert policy and focuses on mimicking this behavior. This approach treats the expert policy as a black box, learning to replicate its decisions without necessarily understanding the underlying motivations. In IL, the reliance on expert demonstrations aligns closely with the concept of an active learning (AL) oracle discussed in Sec. 4.4, which provides guidance in the form of expert input, albeit in a more passive manner.

IL has been effectively applied in robotic scenarios, including indoor autonomous exploration, path planning, and multi-robot coordination. Liu et al. [46] introduce the Learning to Explore (L2E) framework, combining IL with deep RL for UGVs in indoor environments. This approach pretrains exploration policies using IL and then refines them with RL, leading to improved sample efficiency and training speed and outperforming heuristic methods in diverse environments.

Choudhury et al. [31] propose a data-driven IL framework to train planning policies by imitating a clairvoyant oracle with ground truth knowledge of the world map. This approach efficiently trains policies based on partial information for tasks like AIPP and motion planning, showing significant performance gains over state-of-the-art algorithms, even in real UAV applications. Reinhart et al. [42] develop a learning-based path planning approach for UAVs in unknown subterranean environments. They use a graph-based path planner as a training expert for IL, resulting in a policy that guides autonomous exploration with reduced computational costs and less reliance on consistent, online reconstructed maps.

Li et al. [106] address decentralized multi-robot path planning by proposing a combined model that synthesizes local communication and decision-making policies. Their architecture, comprising a convolutional neural network and a graph neural network, was trained to imitate an expert. This model demonstrates its effectiveness in decentralized planning with local communication and observations in cluttered workspaces. Tzes et al. [74] tackles multi-robot AIPP with their Information-aware Graph Block Network (I-GBNet). Trained via IL, this network aggregates information over a communication graph and offers sequential decision-making in a distributed manner. Its scalability and robustness were validated in large-scale experiments involving dynamic target tracking and localization. Lastly, Dai et al. [55] explores indoor environments using an RGB-D camera and generative adversarial imitation learning (GAIL). They modified the ORB-SLAM2 method for enhanced navigation and trained a camera view planning policy with GAIL, yielding efficient exploration and reducing tracking failure in indoor environments.

These studies show the versatility and effectiveness of IL in a wide range of robotic applications, demonstrating its potential to enhance efficiency, decision-making, and adaptability in varied and challenging environments. However, significant challenges and areas for future research remain. Key challenges include improving the generalizability of learned behaviors to unseen environments, enhancing the robustness of IL algorithms against dynamic and unpredictable real-world conditions, and developing more efficient methods for collecting and using expert demonstrations. Further work is also needed to integrate IL with other learning paradigms, such as RL, to address complex multi-agent systems and high-dimensional state spaces.

4.4. Active Learning

The problem of AIPP can also be seen as an active learning (AL) process, whereby the learner uses information measures to choose actions for collecting useful or descriptive data [15]. Here, we mathematically connect the general concept of AL to robotic AIPP theory. We then draw parallels between different components of AL and the learning-based AIPP methods presented in the previous subsections. Lastly, we elaborate on an emerging line of work in AIPP that uses AL objectives to improve sensor models in new environments.

AL addresses the task of finding the most informative data in a set of unlabeled samples such that a model is maximally improved if that sample is labeled and added to the existing training dataset [132]. We define the AL problem as follows:

- $\mathcal{Q}$ is the space of *queries* $q_i \in \mathcal{Q}$ and $N_q \in \mathbb{N}$ is the number of sampled queries;
- $\mathcal{L}$ is the space of *targets* $l_i \in \mathcal{L}$ corresponding to query q_i;
- $\theta : \mathcal{Q} \to \mathcal{L}$ is the *model* predicting a target $\theta(q_i) = l_i$ given a query q_i;
- $P : \Theta \to \mathbb{R}$ is the *performance function* evaluating a model $\theta \in \Theta$;
- $O : \mathcal{Q} \to \mathcal{L}$ is an *oracle* that can be queried to generate a target $O(q_i) = l_i$ for a specific query q_i;
- $q = (q_1, \dots, q_i) \in \mathcal{Q}^i$, $l = (O(q_1), \dots, O(q_i)) \in \mathcal{L}^i$ are collected queries and targets up to iteration $i \leq N_q$;
- $A : \Theta \times \mathcal{Q} \to \mathbb{R}$ is the *acquisition function* estimating a query's q_i effect on the model performance $P(\theta)$ after re-training on collected training data pairs q and l.

We aim to maximize the model's performance with a low number of sampled queries answered by the oracle. Formally, AL solves the following optimization problem:

$$q^* = \underset{q \in \mathcal{Q}^{N_q}}{\operatorname{argmax}} \sum_{i=1}^{N_q} A(\theta, q_i)\,. \tag{5}$$

The AIPP problem in Eq. (1) and AL in Eq. (5) are closely related. In the AL setup, the costs $C(q) = |q|$ of executing a sequence of queries q is given by the number of queries $|q|$ in q. The optimization constraints of Eq. (5) are implicitly determined by $C(q) \leq N_q$. The AIPP information criterion I translates to $I(q) = \sum_{i=1}^{N_q} A(\theta, q_i)$ by substituting the AL acquisition function as the mission objective.

In the classical environmental monitoring or exploration problem setup, the environment ξ is characterized by the set of available queries $\mathcal{Q}$, e.g., robot poses $q_i = \mathbf{x}_i \in \xi$. The feature space $\mathcal{F}$ encompasses the set of targets $\mathcal{L}$ establishing a correspondence between feature mappings $F : \xi \times \mathbb{R} \rightarrow \mathcal{F}$ and the model θ. Further, we cannot directly quantify the true effect of a sequence of queries q on the model performance $P(\theta)$ as this requires re-training on not-yet-acquired oracle-generated targets. Instead, similar to the belief update of $\tilde{F}$ over the true feature mapping F in Eq. (2), at each iteration i, AL methods leverage already collected queries $\{q_1, \ldots, q_i\}$ and targets $\{O(q_1), \ldots, O(q_i)\}$ to update the model θ and ensure maximally informed sequential query selection. We establish this theoretical connection between AIPP and the AL problem to show that the AIPP methods introduced in Sec. 4.1 to 4.3 can be viewed as methods for AL in the context of embodied robotic applications. In the following, we identify the *models*, *queries*, *oracles*, and *acquisition functions* used in different learning-based approaches from an AL perspective.

The *model* could have any parametric or non-parametric form. As discussed in Sec. 4.1, common models for learning a spatio(temporal) representation F of a feature space $\mathcal{F}$ in an environment ξ are non-parametric Gaussian processes or implicit neural representations to probabilistically model $O : \xi \rightarrow \mathcal{F}$. In this case, a *query* q_i is defined by a specific location or area in the environment $q_i = \mathbf{x}_i \in \xi$, and the *oracle*'s answer $O(\mathbf{x}_i) = z_i$ is typically given by collecting a new potentially noisy measurement $z_i \in \mathcal{F}$ at the location $\mathbf{x}_i$ based on the sensor readings. As summarized in Table 2, the performance metric P, i.e., evaluation metric, and thus *acquisition function* A, i.e., information criterion I, varies with the task. Reductions in the covariance matrix trace of a Gaussian process model or model uncertainty of an implicit neural representation are popular acquisition functions used in exploration and monitoring tasks. Approaches leveraging a well-defined probabilistic map belief as a *model* and maximising an information-theoretically motivated *acquisition function* are often closely related to the theory of optimal experimental design [133]. Examples are AIPP approaches that sequentially update Gaussian process map beliefs aiming to maximize its covariance trace reduction [2, 3, 62, 65, 71] as they are derived from A-optimal experiment design strategies. Other supervised approaches learn the acquisition function A with deep neural networks [28, 56, 107] or directly predict the next best query q^*_{i+1}, e.g., a view pose to reconstruct a 3D object [47, 48].

Similarly, RL-based AIPP approaches presented in Sec. 4.2 rely on a reward function $R : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$ based on the current and next state and an action $a_i \in \mathcal{A}$ to learn a policy that predicts the action $a^*_{i+1} \in \mathcal{A}$ maximizing the return, i.e., the sum of rewards. According to the definitions of the return in Eq. (3) and the AL problem in Eq. (5), the reward function in RL acts as the acquisition function in AL. In a POMDP describing the AIPP problem, the state s can be seen as the current model θ trained on the history of collected queries $\{q_1, \ldots, q_i\}$ and targets $\{O(q_1), \ldots, O(q_i)\}$. The queries $q_i = a_i \in \mathcal{A}$ correspond to the previously chosen actions a_i, e.g., the next measurement location, and the oracle-generated targets $O(q_i) = z_i \in \mathcal{O}$ correspond to the observations, e.g., sensor readings at measurement locations.

The AIPP approaches described so far assume static sensor models $p(z \,|\, F)$ to learn spatial environment models $\tilde{F}$ in Eq. (2). However, as environments ξ are initially unknown, sensor models should be adapted to the robot environment to improve robot perception and, thus, the learned environment model $\tilde{F}$. This is crucial for continuous robot deployment in varying domains and for deep learning-based sensor models as they are known to transfer poorly to unseen scenarios.

To address this problem, a line of recent research considers using AIPP to actively gather data to improve a sensor model for robotic perception [88–90, 92, 134]. In this problem setup, the information criterion I in Eq. (1) is defined in terms of how potential future raw sensor readings, e.g., RGB images at certain environment locations, and their associated oracle-generated targets, e.g., semantic segmentation labels, impact the sensor performance. To solve this problem, the key idea is to link the AIPP objective to ideas from AL.

To date, this problem has only been studied in the context of pixel-wise semantic segmentation from RGB images [88–90, 92, 134]. These approaches can be divided into two categories depending on how they implement the oracle for labeling new data: self-supervised methods exploit pseudo labels rendered from an online-built semantic map as an oracle, without considering a human annotator, while fully supervised methods only exploit dense pixel-wise human annotations as an oracle. In the first category, self-supervised methods can be used to fuse 2D semantic predictions from different viewpoints into a semantic 3D map [89, 92]. After a mapping mission is completed, the semantic 3D map is used to automatically render consistent 2D pseudo labels for re-training. In this line of work, Zurbrügg et al. [89] propose an AIPP approach to plan viewpoints with high training data novelty for improved pseudo label quality. Similarly, Chaplot et al. [92] train an RL agent to target low-confidence parts of a 3D map. Although these approaches do not require a human oracle [89, 92], they still rely on large labeled pre-training datasets to generate high-quality pseudo labels in new scenes. In contrast, fully supervised methods are more applicable to varying domains and environments. Blum et al. [88] locally plan paths of high training data novelty. Rückin et al. [134] introduce a map-based planning framework supporting multiple acquisition functions [90]. However, these methods [88, 90, 134] require considerable human labeling effort to improve robotic vision.

These works highlight the versatility of applying AL methods to robotic learning through AIPP. The combination of AL and AIPP methods demonstrates a principled approach to integrating learning-based components and classically used uncertainty quantification techniques for AL. This integration represents a promising pathway towards robot deployment in more diverse environments with minimal human supervision. However, to harness the potential of AL and AIPP, future research must address multiple open problems. Existing uncertainty quantification methods for deep learning models are known to be overconfident [135] and not well-calibrated in out-of-distribution scenarios. Further, fully and self-supervised approaches for sensor model improvement are limited in terms of human annotation effort and applicability to out-of-distribution environments, respectively. One possible direction is to take the strengths of both approaches and develop hybrid semi-supervised approaches [91]. We may be able to deploy more versatile robots by combining AIPP methods for environment monitoring and exploration (Sec. 4.1 to Sec. 4.3) with methods for targeted training data collection [88–90, 92, 134].

5. Applications

In this section, we categorize the AIPP works surveyed in our paper according to their target applications in robotics. Our goal is to assess the practical applicability of existing learning-based methods while pinpointing emerging trends and recognizing potential limitations.

Table 5 overviews AIPP applications in robotics and describes their relevant practical aspects. Our taxonomy considers four broad application areas: (i) environmental monitoring; (ii) exploration and search; (iii) semantic scene understanding; and (iv) active SLAM. For reference, in the last column, we also include the learning method: supervised learning ('SL'); reinforcement learning ('RL'); imitation learning ('IL'); and/or active learning ('AL'). In addition, in Fig. 3, we provide visual summary statistics for our survey according to the application area, learning method, and planning space considered by each paper.

In general, our survey illustrates that the works are broadly applied in terrestrial, aquatic, and aerial domains and on diverse robot platforms. AIPP methods organically find usage in environmental monitoring scenarios to gather data about physical parameters, e.g., signal strength [4, 59, 60, 63, 68, 120], temperature [5, 27, 76], or elevation [6, 45]. In particular, AIPP enables efficient data collection in large environments using sensors with limited coverage area, such as point-based sensors. For mapping, Gaussian processes are most commonly used to capture spatial correlations found in natural phenomena that are measured by a continuous variable. Several recent works use RL to learn the best data-gathering actions [2, 5, 59], including achieving coordinated monitoring with multi-robot teams [27, 64, 120, 136].

AIPP strategies for exploration and search can be further broken down in terms of two distinct tasks: exploration of bounded unknown environments, i.e., outwards-facing scenarios, and active reconstruction of unknown objects or scenes, i.e., inwards-facing scenarios. The former necessarily accounts for possible obstacles in unknown regions and typically relies on depth-based sensing to construct volumetric maps as an environment is explored [31, 32, 41, 54, 74]. In contrast, most approaches for object reconstruction [47, 48, 77, 81, 84] assume obstacle-free environments with a constrained hemispherical action space for a robot camera around the target object of interest. While NeRF-based methods are gaining popularity for this task [77, 78, 81–84, 97], an open research challenge is applying them for exploration in more complex environments involving multiple objects and clutter.

Only a few works integrate semantic information or consider localization uncertainty in planning objectives for learning-based AIPP. Several approaches [88–91, 94, 134] leverage semantics to improve deep learning-based sensor models via AL. Although semantic scene understanding provides valuable cues for learning complex AIPP behavior [38], a key development barrier is the lack of high-quality semantic datasets relevant to the domains of robotic AIPP applications. The common assumption of perfect localization uncertainty represents a significant limitation to the applicability of existing methods, particularly in outdoor environments which may rely on imprecise Global Positioning System (GPS) signals [7, 27, 49, 65, 72]. Many approaches also neglect sensor uncertainty and noise, especially in environmental monitoring scenarios where models for specialized sensors are scarce. This is a major drawback since imperfect sensing has been shown to have significant effects on the quality of collected data and resulting planning strategy [3, 45, 126].

Our taxonomy reveals a gap in methods accounting for robot motion constraints while planning, with some methods assuming that the robot sensor can teleport between locations [77, 78, 81] or perform only incremental motions relative to its current position [27, 35, 120]. One open issue is the lack of guarantees on dynamic feasibility, which is essential for long-term path planning with agile, fast platforms such as small UAVs. Although there has been research on multi-robot missions [27, 53, 63, 72, 136], there are several areas that have not been widely studied, such as tasks involving multi-sensor fusion [62], temporal changes

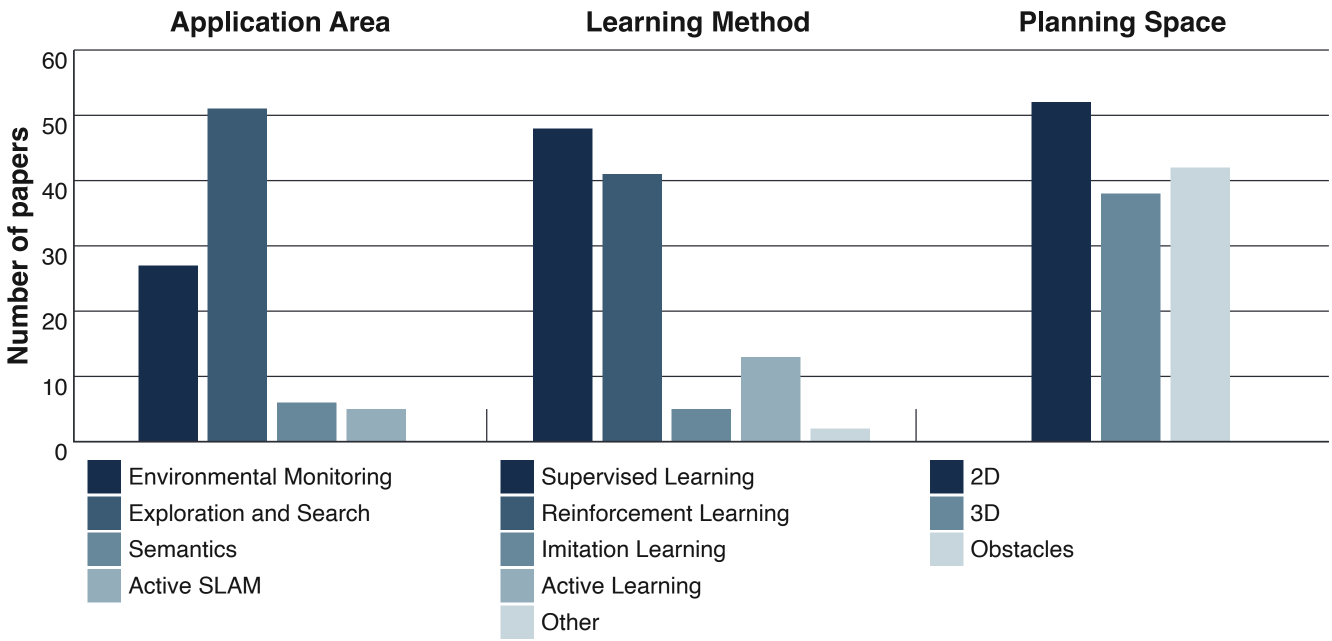

Fig 3: Summary statistics for the papers reviewed in Sec. 5.

in the environment [49, 71], and combining multiple problems simultaneously, e.g., semantic scene understanding and active SLAM.

		Domain	Robot platforms	Sensors	Obstacles?	Planning space	Sources of uncertainty	Motion constraints	Learning method
Environmental monitoring	[95]	Air quality monitoring	UAV	Point-based gas sensor	✗	2D	Mapping	-	-
	[69]	Air quality monitoring	UGV	Point-based sensor	✗	2D	Mapping	-	SL, RL
	[6]	Elevation mapping	UAV team	Ultrasound height sensor	✓	2D	Mapping	-	SL, RL
	[43]	Fruit monitoring	Robot arm	RGB-D camera	✓	3D	Mapping	ROS MoveIt*	RL
	[137]	Fruit monitoring	Robot arm	RGB-D camera	✓	3D	Mapping	ROS MoveIt*	SL
	[59]	Intensity monitoring	AUV/UGV/UAV	Generic camera, point-based sensor	✗	2D/3D	Mapping	-	SL, RL
	[2]	Intensity monitoring	UGV	Point-based sensor	✗	2D	Mapping	-	SL, RL
	[63]	Intensity monitoring	UGV team	Point-based sensor	✗	2D	Mapping	-	SL, RL
	[60]	Intensity monitoring	UAV	Point-based sensor	✗	2D	Mapping, sensing	Nonholonomic constraints	SL
	[65]	Lake bacteria monitoring	AUV	Point sensor for phycoerythrin fluorescence	✗	3D	Mapping	Spline path	SL
	[73]	Ocean pollution monitoring	AUV	Pressure sensor, general water characterization	✗	2D	Mapping	Spline path	RL
	[3]	Precision agriculture	UAV	Generic camera	✗	3D	Mapping, sensing, semantics	Spline path	SL
	[5]	Precision agriculture	UAV	Generic camera	✗	3D	Mapping, sensing	-	SL, RL
	[27]	Precision agriculture	UAV team	Generic camera	✗	3D	Mapping, sensing	-	RL
	[126]	Scientific data gathering	UGV	Generic camera, ultraviolet light source, laser scanner	✗	2D	Mapping, sensing	Omnidirectional drive	RL
	[68]	Signal strength monitoring	AUV	Point-based sensor	✗	2D	Mapping	-	SL
	[4]	Signal strength monitoring	AUV/UGV	Generic camera, point-based sensor	✓	2D	Mapping, path executability	-	RL
	[66]	Signal strength monitoring	UGV	Point-based sensor	✗	2D	Mapping	-	SL, RL
	[49]	Spatiotemporal field monitoring	AUV/UGV	Point-based sensor	✗	2D	Mapping	-	SL
	[71]	Spatiotemporal field monitoring	UGV	Point-based sensor	✗	2D	Mapping	Spline path	SL, RL
	[74]	Target tracking and localization	UGV team	RGB-D sensor	✗	2D	Mapping, sensing	-	IL
	[76]	Temperature monitoring	Sensor network	Generic	✗	2D	Mapping	-	SL
	[67]	Tree disease classification	UAV	RGB camera	✗	2D	Mapping, semantics	-	SL
	[45]	Underwater mapping	AUV	Sonar	✗	3D	Mapping, sensing	Dubins vehicle model	SL
	[64]	Water quality monitoring	AUV team	Water quality sensor	✗	2D	Mapping	-	SL, RL
	[35]	Wildfire monitoring	UAV team	Thermal camera	✗	2D	-	-	RL
	[120]	Wireless data harvesting	UAV team	Signal receptor	✓	2D	Signal emissions	-	RL
Exploration and search	[55]	Camera view planning	UGV	RGB-D camera	✓	3D	-	-	IL
	[121]	Exploration and landmark localization	UAV	Depth sensor	✗	2D/3D	Mapping	-	RL
	[136]	Exploration and target search	UGV team	Depth sensor	✓	2D	-	-	RL
	[31]	Exploration, search-based planning	UGV/UAV	RGB-D camera	✓	2D	Mapping	-	IL
	[32]	Floor plan exploration	UAV	Laser	✓	2D	Mapping, sensing	Unicycle model	RL
	[46]	Indoor exploration	UGV	Depth sensor	✓	2D	-	-	RL, IL
	[54]	Indoor exploration	UGV	Depth sensor	✓	2D	Map prediction	-	SL, RL
	[28]	Indoor exploration	UGV	Depth sensor	✓	2D	Mapping, sensing	-	SL
	[51]	Indoor exploration	UAV	Depth sensor	✓	3D	Mapping, sensing	Spline path	SL
	[41]	Indoor exploration	UGV	Depth sensor	✓	2D	Mapping, sensing	Local planner* [138]	RL

* Motion constraints incorporated only during plan execution.

Table 5: Taxonomy of AIPP applications in robotics.

		Domain	Robot platforms	Sensors	Obstacles?	Planning space	Sources of uncertainty	Motion constraints	Learning method
Exploration and search	[40]	Indoor exploration	UGV	Depth sensor	✓	2D	Mapping, sensing	ROS move_base*	RL
	[33]	Indoor exploration	UGV	Depth sensor	✓	2D/3D	Mapping, sensing	-	RL
	[44]	Indoor exploration	UGV	Laser scanner	✓	2D	Mapping	-	SL
	[52]	Indoor exploration	UGV	RGB-D camera	✓	2D	-	Local planner using Fast Marching Method	SL
	[39]	Indoor exploration	UGV	RGB-D camera	✓	2D	Mapping, sensing	-	SL
	[38]	Indoor exploration	UAV	RGB-D camera	✓	3D	Mapping, sensing, scene completion	Velocity ramp model	SL
	[36]	Indoor exploration, point-goal navigation	UGV/UAV	Depth sensor	✓	2D	Mapping, sensing	-	SL
	[37]	Indoor exploration	UGV/UAV	RGB-D camera	✓	2D	Mapping, sensing	-	SL
	[93]	Infrastructure inspection	UAV	RGB-D camera	✗	3D	-	-	RL
	[98]	Interactive exploration	Robot arm	RGB-D camera	✗	3D	-	-	RL
	[50]	Multi-object navigation	UGV	RGB-D camera	✓	2D	-	Revolute joint camera	RL
	[139]	Object finding	Robot arm	RGB-D camera	✓	3D	Mapping, sensing	-	RL
	[75]	Object recognition, team orienteering	UGV team	2D Laser	✓	2D	Mapping, sensing	-	RL
	[48]	Object reconstruction	Any platform	Depth sensor	✗	3D	-	-	SL
	[57]	Object reconstruction	Any platform	Depth sensor	✓	3D	-	-	SL
	[47]	Object reconstruction	Any platform	Depth sensor	✗	3D	Mapping	-	SL
	[82]	Object reconstruction	Any platform	RGB camera	✗	3D	-	-	SL
	[78]	Object reconstruction	Any platform	RGB camera	✗	3D	Sensing	-	SL, AL
	[80]	Object reconstruction	Any platform	RGB camera	✗	3D	Sensing	-	SL, AL
	[81]	Object reconstruction	Any platform	RGB camera	✗	3D	Sensing	-	SL, AL
	[77]	Object reconstruction	Any platform	RGB camera	✗	3D	Sensing	-	SL, AL
	[30]	Object reconstruction	Robot arm	RGB-D camera	✓	3D	Mapping, sensing	-	RL
	[84]	Object reconstruction	Any platform	RGB-D camera	✗	3D	Sensing	-	SL, AL
	[79]	Object reconstruction	Any platform	RGB-D camera	✗	3D	Sensing	-	SL, AL
	[83]	Object reconstruction	Any platform	RGB-D camera	✓	3D	Sensing	-	SL, AL
	[57]	Object reconstruction	UAV	Depth sensor	✓	3D	-	ROS MoveIt*	SL
	[58]	Object surface reconstruction	Any platform	RGB-D camera	✗	3D	Sensing	ROS MoveIt*	SL
	[70]	Persistent target monitoring	Robot team (UAV/UGV)	Generic camera	✗	2D	Target position	-	SL, RL
	[62]	Rover exploration	UGV	Drill, mass spectrometer	✓	2D	Mapping, sensing	-	RL
	[7]	Search and rescue	Any platform	Categorical survivor detection	✓	2D	Mapping, sensing	-	RL
	[8]	Search and rescue	Any platform	Optical sensor	✗	2D	Sensing	-	RL
	[104]	Search tasks, e.g., grasping pose or target search	Any platform	Binary measurements	✗	3D	Sensing	-	-
	[42]	Subterranean exploration	UAV	Depth sensor	✓	3D	-	-	IL
	[124]	Target assignment	UAV team	Not specified	✗	2D	-	-	RL
	[53]	Target assignment	UAV team	Not specified	✗	2D	Mapping, sensing	-	RL
	[72]	Target tracking	UAV team	Laser	✗	2D	Mapping	Unicycle model	RL
	[130]	Targeted object search	UGV	RGB camera	✓	2D	Object recognition	-	RL
	[96]	Tunnel exploration	UGV	Depth sensor	✓	2D	-	-	SL
	[107]	Urban exploration, surveillance	UGV/UAV	Depth sensor	✓	2D	-	-	SL
	[56]	Urban exploration	UAV	Depth sensor	✓	3D	Mapping, sensing	-	SL
	[97]	View planning for grasping	Any platform	RGB-D camera	✗	3D	-	-	SL, AL

		Domain	Robot platforms	Sensors	Obstacles?	Planning space	Sources of uncertainty	Motion constraints	Learning method
Semantics	[88]	Land cover mapping	UAV	RGB camera	✗	2D	Sensing	-	AL
	[94]	Indoor goal navigation	UGV	RGB-D camera	✓	2D	Mapping, sensing	-	AL
	[89]	Indoor semantic mapping	UGV	Laser, RGB-D camera	✓	3D	Mapping, sensing, localization	-	AL
	[90]	Industrial and urban mapping, land cover analysis	UAV	RGB camera	✗	2D	Mapping, sensing	-	
	[134]	Urban mapping	UAV	RGB camera	✗	2D	Mapping, sensing	-	AL
	[91]	Urban mapping	UAV	RGB camera	✗	2D	Mapping, sensing	-	AL
Active SLAM	[108]	Active visual localization	UGV	RGB-D camera	✓	3D	Localization	-	SL
	[61]	Indoor temperature monitoring	UGV/UAV	Laser, temperature sensor	✓	2D/3D	Mapping, sensing, localization	-	SL
	[29]	Exploration	UGV	Lidar	✗	2D	Mapping, sensing, localization	-	RL
	[34]	Exploration	UGV	Lidar	✓	2D	Mapping, sensing, localization	-	RL
	[122]	Goal-based navigation	UAV	RGB-D camera	✓	3D	Mapping, sensing, localization	Spline path	RL

6. Challenges and Future Directions

This section outlines the open challenges related to deploying learning-based AIPP in robotics settings and avenues for future work to overcome them.

6.1. Generalizability

Most of the AIPP methods surveyed were tested in the same or similar environments as those seen during training. The generalizability of deep learning-based systems is especially important because the neural networks that are often used in AIPP frameworks [31, 38, 44, 51, 55, 106] are known to produce overconfident wrong predictions in out-of-distribution scenarios [135]. Future work should address these inherent generalization limitations to improve robustness. We identified one possible way to deal with deep learning-based vision uncertainties in Sec. 4.4 by combining uncertainty-aware deep learning and active learning approaches with AIPP [89, 90, 92]. Similarly, RL-based AIPP systems are increasingly popular [2, 34, 41, 63, 66, 69]. However, RL approaches degrade in performance when deployed in unseen simulated environments or real-world scenarios [140–142]. Future work on RL-based AIPP methods should focus on evaluating the learned policy in more diverse environments and in the real world. Further, new methods are needed to improve the generalization of RL for AIPP [143]. Other subfields of RL, such as meta-reinforcement learning [144], could serve as inspiration for enabling efficient online policy adaption to unseen environments. Data augmentation and domain randomization during training could improve test-time policy robustness [145].

6.2. Handling Localization Uncertainty

Effectively handling robot localization uncertainty is a key aspect in advancing learning-based AIPP in robotics. Localization uncertainties arise from various sources, including sensor noise, environmental changes, and the inherent limitations of localization technologies. However, most existing approaches assume perfect knowledge of the robot pose, both during planning and plan execution, which limits their practical applicability. Some works in active SLAM account for localization uncertainty via belief space planning [29, 34] or modifying the planning objective [61] using graph-based representations of the robot state. Incorporating such probabilistic methods into learning-based AIPP methods and exploring transfer learning techniques [34, 146] can enhance robustness.

6.3. Temporal Changes and Long-Term Planning

Addressing temporal changes and ensuring robust long-term planning in learning-based AIPP poses a critical challenge. Temporal changes, such as variations in environmental conditions, seasonal shifts, or the presence of dynamic obstacles, require AIPP methods to not only adapt to the immediate surroundings but also plan for sustained effectiveness over long durations. This involves developing strategies that consider the evolution of the environment and adjust the planned paths online. Whereas several works study spatiotemporal monitoring [49, 70, 71] and dynamic target tracking [72, 74], there is a need for new approaches that can handle multiple moving objects in 3D environments. We also highlight the importance of developing such methods for agile platforms such as small UAV platforms while also accounting for their physical motion constraints, given that they often operate in dynamic and changing landscapes.

6.4. Heterogeneity

One challenge is the development of learning-based AIPP methods that exploit the benefits of heterogeneous robot teams or heterogeneous sensor modalities within a single robot. Leveraging heterogeneity in robotic systems has a high potential to improve sensing and monitoring performance. For instance, a tandem robotic system can combine the agility of a small UAV and the high-payload, detailed inspection capabilities of a UGV. A robot equipped with different sensors, e.g., a drill and a mass spectrometer [62], can measure different physical parameters during a single mission. To fully exploit these capabilities, we recommend investigating principled ways of incorporating heterogeneity, also considering collaboration aspects for effective decision-making.

6.5. Standardized Evaluation Methods

Another key challenge is the lack of standardized simulation environments, evaluation metrics, and benchmarks to assess new methods in AIPP. As highlighted in Sec. 3, most existing works rely on custom-made procedures and baselines for evaluation. While these reveal relative trends in performance, they do not ensure reproducible results across different approaches. Moreover, our survey catalogue[1] indicates that very few works have publicly available datasets, documentation, and open-source code. In future work, it is essential to establish common evaluation pipelines to promote reproducibility and comparability. We recommend constructing such a framework in a modular fashion, allowing easy modification of various aspects, including the application scenario (robot platform, sensor setup, and environment) and algorithmic components (mapping method, planning objective, and planning algorithm). We believe such a development would also help guide future work towards more general approaches, rather than constrained application-specific methods, as well as foster stronger collaboration within the AIPP community.

7. Conclusion

In this survey paper, we analyzed the current research on AIPP in robotic applications, focusing especially on

[1] https://dmar-bonn.github.io/aipp-survey/

methods that exploit recent advances in robot learning. We introduced a unified mathematical problem definition for addressing tasks in learning-based AIPP. Based on this formulation, we provided two complementary taxonomies from the perspectives of (i) learning algorithms and (ii) robotic applications. We provided a structured outline of the research landscape and highlighted synergies and trends. Finally, we discussed the remaining challenges for learning-based AIPP in robotics and outlined promising research directions.

8. Funding Sources

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2070 – 390732324.

We would like to thank Jonas Schaap for helping with the infographics design.

References

[1] I. M. Rayas Fernández, C. E. Denniston, D. A. Caron, and G. S. Sukhatme, "Informative Path Planning to Estimate Quantiles for Environmental Analysis," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 4, pp. 10 280–10 287, 2022.

[2] Y. Cao, Y. Wang, A. Vashisth, H. Fan, and G. A. Sartoretti, "CAtNIPP: Context-Aware Attention-based Network for Informative Path Planning," in *Proc. of the Conf. on Robot Learning (CoRL)*, 2023.

[3] M. Popović, T. Vidal-Calleja, G. Hitz, J. J. Chung, I. Sa, R. Siegwart, and J. Nieto, "An informative path planning framework for UAV-based terrain monitoring," *Autonomous Robots*, vol. 44, pp. 889–911, 2020.

[4] C. E. Denniston, O. Peltzer, J. Ott, S. Moon, S.-K. Kim, G. S. Sukhatme, M. J. Kochenderfer, M. Schwager, and A.-a. Aghamohammadi, "Fast and Scalable Signal Inference for Active Robotic Source Seeking," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[5] J. Rückin, L. Jin, and M. Popović, "Adaptive Informative Path Planning Using Deep Reinforcement Learning for UAV-based Active Sensing," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[6] A. Viseras and R. Garcia, "DeepIG: Multi-robot information gathering with deep reinforcement learning," *IEEE Robotics and Automation Letters (RA-L)*, vol. 4, no. 3, pp. 3059–3066, 2019.

[7] S. Choudhury, N. Gruver, and M. J. Kochenderfer, "Adaptive Informative Path Planning with Multimodal Sensing," in *International Conference on Automated Planning and Scheduling (ICAPS)*, 2020.

[8] A. Singh, A. Krause, and W. J. Kaiser, "Nonmyopic Adaptive Informative Path Planning for Multiple Robots," in *Proc. of the Intl. Conf. on Artificial Intelligence (IJCAI)*, 2009.

[9] E. Galceran and M. Carreras, "A Survey on Coverage Path Planning for Robotics," *Journal on Robotics and Autonomous Systems (RAS)*, vol. 61, no. 12, pp. 1258–1276, 2013.

[10] C. S. Tan, R. Mohd-Mokhtar, and M. R. Arshad, "A Comprehensive Review of Coverage Path Planning in Robotics Using Classical and Heuristic Algorithms," *IEEE Access*, vol. 9, pp. 119 310–119 342, 2021.

[11] S. Bai, T. Shan, F. Chen, L. Liu, and B. Englot, "Information-Driven Path Planning," *Current Robotics Reports*, no. 2, pp. 177–188, 2021.

[12] M. Maboudi, M. Homaei, S. Song, S. Malihi, M. Saadatseresht, and M. Gerke, "A Review on Viewpoints and Path Planning for UAV-Based 3D Reconstruction," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2023.

[13] I. Lluvia, E. Lazkano, and A. Ansuategi, "Active Mapping and Robot Exploration: A Survey," *Sensors*, vol. 21, no. 7, 2021.

[14] Y. Sung, J. Das, and P. Tokekar, "Decision-Theoretic Approaches for Robotic Environmental Monitoring – A Survey," *arXiv preprint arXiv:2308.02698*, 2023.

[15] A. T. Taylor, T. A. Berrueta, and T. D. Murphey, "Active learning in robotics: A review of control principles," *Mechatronics*, vol. 77, p. 102576, 2021.

[16] M. Aniceto and K. C. T. Vivaldini, "A Review of the Informative Path Planning, Autonomous Exploration and Route Planning Using UAV in Environment Monitoring," in *Intl. Conf. on Computational Science and Computational Intelligence (CSCI)*, 2022.

[17] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and autonomous systems*, vol. 57, no. 5, pp. 469–483, 2009.

[18] L. C. Garaffa, M. Basso, A. A. Konzen, and E. P. de Freitas, "Reinforcement Learning for Mobile Robotics Exploration: A Survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 8, pp. 3796–3810, 2023.

[19] M. Lauri, D. Hsu, and J. Pajarinen, "Partially observable Markov decision processes in robotics: A survey," *IEEE Trans. on Robotics (TRO)*, vol. 39, no. 1, pp. 21–40, 2022.

[20] D. Mukherjee, K. Gupta, L. H. Chang, and H. Najjaran, "A survey of robot learning strategies for human-robot collaboration in industrial settings," *Robotics and Computer-Integrated Manufacturing*, vol. 73, p. 102231, 2022.

[21] S. Chen, "Kalman filter for robot vision: a survey," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4409–4420, 2011.

[22] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.

[23] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning.* MIT Press, 2006.

[24] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[25] J. A. Placed, J. Strader, H. Carrillo, N. Atanasov, V. Indelman, L. Carlone, and J. A. Castellanos, "A Survey on Active Simultaneous Localization and Mapping: State of the Art and New Frontiers," *IEEE Trans. on Robotics (TRO)*, 2023.

[26] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, "pixelNeRF: Neural radiance fields from one or few images," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[27] J. Westheider, J. Rückin, and M. Popović, "Multi-UAV Adaptive Path Planning Using Deep Reinforcement Learning," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

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

[29] F. Chen, J. D. Martin, Y. Huang, J. Wang, and B. Englot, "Autonomous Exploration Under Uncertainty via Deep Reinforcement Learning on Graphs," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[30] F. Sukkar, G. Best, C. Yoo, and R. Fitch, "Multi-robot region-of-interest reconstruction with dec-mcts," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2019.

[31] S. Choudhury, M. Bhardwaj, A. Kapoor, G. Ranade, S. Scherer, and D. Dey, "Data-driven planning via imitation learning," *Intl. Journal of Robotics Research (IJRR)*, vol. 37, pp. 1632–1672, 2018.

[32] M. Lodel, B. Brito, A. Serra-Gómez, L. Ferranti, R. Babuška,

and J. Alonso-Mora, "Where to Look Next: Learning Viewpoint Recommendations for Informative Trajectory Planning," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[33] F. Chen, S. Bai, T. Shan, and B. Englot, "Self-Learning Exploration and Mapping for Mobile Robots via Deep Reinforcement Learning," in *AIAA SciTech Forum*, 2019.

[34] F. Chen, P. Szenher, Y. Huang, J. Wang, T. Shan, S. Bai, and B. Englot, "Zero-Shot Reinforcement Learning on Graphs for Autonomous Exploration Under Uncertainty," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2021.

[35] A. Viseras, M. Meissner, and J. Marchal, "Wildfire Front Monitoring with Multiple UAVs using Deep Q-Learning," *IEEE Access*, 2021.

[36] G. Georgakis, B. Bucher, A. Arapin, K. Schmeckpeper, N. Matni, and K. Daniilidis, "Uncertainty-driven Planner for Exploration and Navigation," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[37] S. K. Ramakrishnan, Z. Al-Halah, and K. Grauman, "Occupancy Anticipation for Efficient Exploration and Navigation," in *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2020.

[38] L. Schmid, M. N. Cheema, V. Reijgwart, R. Siegwart, F. Tombari, and C. Cadena, "SC-Explorer: Incremental 3D Scene Completion for Safe and Efficient Exploration Mapping and Planning," *arXiv preprint arXiv:2208.08307*, 2023.

[39] L. Schmid, C. Ni, Y. Zhong, R. Siegwart, and O. Andersson, "Fast and Compute-Efficient Sampling-Based Local Exploration Planning via Distribution Learning," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 3, pp. 7810–7817, 2022.

[40] F. Niroui, K. Zhang, Z. Kashino, and G. Nejat, "Deep Reinforcement Learning Robot for Search and Rescue Applications: Exploration in Unknown Cluttered Environments," *IEEE Robotics and Automation Letters (RA-L)*, vol. 4, no. 2, pp. 610–617, 2019.

[41] Y. Cao, T. Hou, Y. Wang, X. Yi, and S. Sartoretti, "ARiADNE: A Reinforcement learning approach using Attention-based Deep Networks for Exploration," *arXiv preprint arXiv:2301.11575*, 2023.

[42] R. Reinhart, T. Dang, E. Hand, C. Papachristos, and K. Alexis, "Learning-based Path Planning for Autonomous Exploration of Subterranean Environments," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[43] X. Zeng, T. Zaenker, and M. Bennewitz, "Deep Reinforcement Learning for Next-Best-View Planning in Agricultural Applications," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[44] R. Shrestha, F.-P. Tian, W. Feng, P. Tan, and R. Vaughan, "Learned Map Prediction for Enhanced Mobile Robot Exploration," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2019.

[45] L. Zacchini, A. Ridolfi, and B. Allotta, "Informed expansion for informative path planning via online distribution learning," *Journal on Robotics and Autonomous Systems (RAS)*, vol. 166, p. 104449, 2023.

[46] Z. Liu, M. Deshpande, X. Qi, D. Zhao, R. Madhivanan, and A. Sen, "Learning to Explore (L2E): Deep Reinforcement Learning-based Autonomous Exploration for Household Robot," in *Robotics: Science and Systems Workshop on Robot Representations for Scene Understanding, Reasoning, and Planning*, 2023.

[47] J. I. Vasquez-Gomez, D. Troncoso, I. Becerra, E. Sucar, and R. Murrieta-Cid, "Next-best-view regression using a 3D convolutional neural network," *Machine Vision and Applications*, vol. 32, no. 42, 2021.

[48] M. Mendoza, J. I. Vasquez-Gomez, H. Taud, L. E. Sucar, and C. Reta, "Supervised learning of the next-best-view for 3d object reconstruction," *Pattern Recognition Letters*, vol. 133, pp. 224–231, 2020.

[49] J. A. Caley, N. R. J. Lawrance, and G. A. Hollinger, "Deep learning of structured environments for robot search," *Autonomous Robots*, vol. 43, pp. 1695–1714, 2019.

[50] P. Chen, D. Ji, K. Lin, W. Hu, W. Huang, T. Li, M. Tan, and C. Gan, "Learning Active Camera for Multi-Object Navigation," in *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, 2022, pp. 28 670–28 682.

[51] Y. Tao, Y. Wu, B. Li, F. Cladera, A. Zhou, D. Thakur, and V. Kumar, "SEER: Safe Efficient Exploration for Aerial Robots using Learning to Predict Information Gain," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[52] Y. Li, A. Debnath, G. Stein, and J. Kosecka, "Learning-Augmented Model-Based Planning for Visual Exploration," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[53] M. Gao and X. Zhang, "Cooperative Search Method for Multiple UAVs Based on Deep Reinforcement Learning," *Sensors*, vol. 22, no. 18, p. 6737, 2022.

[54] E. Zwecher, E. Iceland, S. R. Levy, S. Y. Hayoun, O. Gal, and A. Barel, "Integrating Deep Reinforcement and Supervised Learning to Expedite Indoor Mapping," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[55] X.-Y. Dai, Q.-H. Meng, S. Jin, and Y.-B. Liu, "Camera view planning based on generative adversarial imitation learning in indoor active exploration," *Applied Soft Computing*, vol. 129, p. 109621, 2022.

[56] B. Hepp, D. Dey, S. N. Sinha, A. Kapoor, N. Joshi, and O. Hilliges, "Learn-to-Score: Efficient 3D Scene Exploration by Predicting View Utility," in *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2018.

[57] H. Dhami, V. D. Sharma, and P. Tokekar, "Pred-NBV: Prediction-guided Next-Best-View Planning for 3D Object Reconstruction," *arXiv preprint arXiv:2304.11465*, 2023.

[58] S. Pan, H. Hu, H. Wei, N. Dengler, T. Zaenker, and M. Bennewitz, "One-Shot View Planning for Fast and Complete Unknown Object Reconstruction," *arXiv preprint arXiv:2304.00910*, 2023.

[59] C. Denniston, G. Salhotra, A. Kangaslahti, D. A. Caron, and G. S. Sukhatme, "Learned Parameter Selection for Robotic Information Gathering," *arXiv preprint arXiv:2303.05022*, 2023.

[60] S. Song, A. Rodriguez, and M. Teodorescu, "Trajectory planning for autonomous nonholonomic vehicles for optimal monitoring of spatial phenomena," in *Proc. of the Intl. Conf. on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2015, pp. 40–49.

[61] M. Popović, T. Vidal-Calleja, J. J. Chung, J. Nieto, and R. Siegwart, "Informative Path Planning for Active Field Mapping under Localization Uncertainty," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[62] J. Ott, E. Balaban, and M. J. Kochenderfer, "Sequential Bayesian optimization for adaptive informative path planning with multimodal sensing," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[63] T. Yang, Y. Cao, and S. Sartoretti, "Intent-based Deep Reinforcement Learning for Multi-agent Informative Path Planning," *arXiv preprint arXiv:2303.05351*, 2023.

[64] S. Yanes Luis, M. Perales Esteve, D. Gutiérrez Reina, and S. Toral Marín, *Deep Reinforcement Learning Applied to Multi-agent Informative Path Planning in Environmental Missions*, 2023, pp. 31–61.

[65] G. Hitz, E. Galceran, M.-E. Garneau, F. Pomerleau, and R. Siegwart, "Adaptive continuous-space informative path planning for online environmental monitoring," *Journal of Field Robotics (JFR)*, vol. 34, no. 8, pp. 1427–1449, 2017.

[66] Y. Wei and R. Zheng, "Informative path planning for mobile sensing with reinforcement learning," in *IEEE Conference on Computer Communications*, 2020, pp. 864–873.

[67] K. C. T. Vivaldini, T. H. Martinelli, V. C. Guizilini, J. R. Souza, M. D. Oliviera, F. T. Ramos, and D. F. Wolf, "UAV route planning for active disease classification," *Autonomous Robots*, vol. 43, pp. 1137–1153, 2019.

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

[69] T. Choi and G. Cielniak, "Adaptive Selection of Informative Path Planning Strategies via Reinforcement Learning," in *Proc. of the Europ. Conf. on Mobile Robotics (ECMR)*, 2021.

[70] Y. Cao, T. Hou, Y. Wang, X. Yi, and S. Sartoretti, "Spatio-Temporal Attention Network for Persistent Monitoring of Multiple Mobile Targets," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[71] R. Marchant, F. Ramos, and S. Sanner, "Sequential Bayesian Optimisation for Spatial-Temporal Monitoring," in *Proc. of the Conf. on Uncertainty in Artificial Intelligence (UAI)*, 2014, pp. 553–562.

[72] M. Hüttenrauch, S. Adrian, G. Neumann *et al.*, "Deep reinforcement learning for swarm systems," *Journal of Machine Learning Research*, vol. 20, no. 54, pp. 1–31, 2019.

[73] D. A. Duecker, B. Mersch, R. C. Hochdahl, and E. Kreuzer, "Embedded Stochastic Field Exploration with Micro Diving Agents using Bayesian Optimization-Guided Tree-Search and GMRFs," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021.

[74] M. Tzes, N. Bousias, E. Chatzipantazis, and G. J. Pappas, "Graph Neural Networks for Multi-Robot Active Information Acquisition," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[75] G. Best, O. Cliff, T. Patten, R. R. Mettu, and R. Fitch, "Dec-mcts: Decentralized planning for multi-robot active perception," *Intl. Journal of Robotics Research (IJRR)*, vol. 38, no. 2-3, pp. 316–337, 2019.

[76] A. Meliou, A. Krause, C. Guestrin, and J. M. Hellerstein, "Nonmyopic informative path planning in spatio-temporal models," in *Proc. of the Conf. on Advancements of Artificial Intelligence (AAAI)*, vol. 10, no. 4, 2007, pp. 16–7.

[77] L. Jin, X. Chen, J. Rückin, and M. Popović, "NeU-NBV: Next Best View Planning Using Uncertainty Estimation in Image-Based Neural Rendering," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[78] X. Pan, Z. Lai, S. Song, and G. Huang, "ActiveNeRF: Learning Where to See with Uncertainty Estimation," in *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2022.

[79] Y. Ran, J. Zeng, S. He, L. Li, Y. Chen, G. Lee, J. Chen, and Q. Ye, "NeurAR: Neural Uncertainty for Autonomous 3D Reconstruction with Implicit Neural Representations," *IEEE Robotics and Automation Letters (RA-L)*, vol. 8, no. 2, pp. 1125–1132, 2022.

[80] H. Zhan, J. Zheng, Y. Xi, I. Reid, and H. Rezatofighi, "ActiveRMAP: Radiance Field for Active Mapping And Planning," *arXiv preprint arXiv:2211.12656*, 2023.

[81] N. Sünderhauf, J. Abou-Chakra, and D. Miller, "Density-aware NeRF Ensembles: Quantifying Predictive Uncertainty in Neural Radiance Fields," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[82] S. Pan, L. Jin, H. Hu, M. Popović, and M. Bennewitz, "How Many Views Are Needed to Reconstruct an Unknown Object Using NeRF?" *arXiv preprint arXiv:2310.00684*, 2023.

[83] D. Yan, J. Liu, F. Quan, H. Chen, and M. Fu, "Active Implicit Object Reconstruction Using Uncertainty-Guided Next-Best-View Optimization," *IEEE Robotics and Automation Letters (RA-L)*, vol. 8, no. 10, pp. 6395–6402, 2023.

[84] S. Lee, L. Chen, J. Wang, A. Liniger, S. Kumar, and F. Yu, "Uncertainty Guided Policy for Active Robotic 3D Reconstruction Using Neural Radiance Fields," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 4, pp. 12 070–12 077, 2022.

[85] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *Intl. Journal of Computer Vision (IJCV)*, vol. 88, no. 2, pp. 303–338, 2010.

[86] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3213–3223.

[87] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning.* MIT press Cambridge, MA, 2006, vol. 2, no. 3.

[88] H. Blum, S. Rohrbach, M. Popovic, L. Bartolomei, and R. Siegwart, "Active learning for uav-based semantic mapping," *arXiv preprint arXiv:1908.11157*, 2019.

[89] R. Zurbrügg, H. Blum, C. Cadena, R. Siegwart, and L. Schmid, "Embodied Active Domain Adaptation for Semantic Segmentation via Informative Path Planning," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 4, pp. 8691–8698, 2022.

[90] J. Rückin, F. Magistri, C. Stachniss, and M. Popović, "An Informative Path Planning Framework for Active Learning in UAV-Based Semantic Mapping," *IEEE Trans. on Robotics (TRO)*, vol. 39, no. 6, pp. 4279–4296, 2023.

[91] J. Rückin, F. Magistri, C. Stachniss, and M. Popović, "Semi-Supervised Active Learning for Semantic Segmentation in Unknown Environments Using Informative Path Planning," *IEEE Robotics and Automation Letters (RA-L)*, pp. 1–8, 2024.

[92] D. S. Chaplot, M. Dalal, S. Gupta, J. Malik, and R. R. Salakhutdinov, "Seal: Self-supervised embodied active learning using exploration and 3d consistency," *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, vol. 34, pp. 13 086–13 098, 2021.

[93] S. H. Gazani, M. Tucsok, I. Mantegh, and H. Najjaran, "Bag of Views: An Appearance-based Approach to Next-Best-View Planning for 3D Reconstruction," *arXiv preprint arXiv:2307.05832*, 2023.

[94] G. Georgakis, B. Bucher, K. Schmeckpeper, S. Singh, and K. Daniilidis, "Learning to Map for Active Semantic Goal Navigation," in *Proc. of the Int. Conf. on Learning Representations (ICLR)*, 2022.

[95] O. Velasco, J. Valente, and A. Y. Mersha, "An adaptive informative path planning algorithm for real-time air quality monitoring using UAVs," in *Proc. of the Intl. Conf. on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2020, pp. 1121–1130.

[96] M. Saroya, G. Best, and G. A. Hollinger, "Online Exploration of Tunnel Networks Leveraging Topological CNN-based World Predictions," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[97] X. Zhang, D. Wang, S. Han, W. Li, B. Zhao, Z. Wang, X. Duan, C. Fang, X. Li, and J. He, "Affordance-Driven Next-Best-View Planning for Robotic Grasping," in *Proc. of the Conf. on Robot Learning (CoRL)*, 2023.

[98] K. N. Kumar, I. Essa, and S. Ha, "Graph-based cluttered scene generation and interactive exploration using deep reinforcement learning," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[99] J. Binney and G. S. Sukhatme, "Branch and bound for informative path planning," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*. IEEE, 2012, pp. 2147–2154.

[100] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Intl. Journal of Robotics Research (IJRR)*, vol. 30, no. 7, pp. 846–894, 2011.

[101] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 1997, pp. 146–151.

[102] M. A. Gelbart, J. Snoek, and R. P. Adams, "Bayesian optimization with unknown constraints," *arXiv preprint arXiv:1403.5607*, 2014.

[103] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evolutionary computation*, vol. 9, no. 2, pp. 159–195, 2001.

[104] Z. W. Lim, D. Hsu, and W. S. Lee, "Adaptive informative path planning in metric spaces," *Intl. Journal of Robotics Research (IJRR)*, vol. 35, no. 5, pp. 585–598, 2016.

[105] S. Isler, R. Sabzevari, J. Delmerico, and D. Scaramuzza, "An information gain formulation for active volumetric 3D reconstruction," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2016.

[106] Q. Li, F. Gama, A. Ribeiro, and A. Prorok, "Graph Neural Networks for Decentralized Multi-Robot Path Planning," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[107] L. Ly and Y.-H. R. Tsai, "Autonomous Exploration, Reconstruction, and Surveillance of 3D Environments Aided by Deep Learning," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2019.

[108] M. Hanlon, B. Sun, M. Pollefeys, and H. Blum, "Active Visual Localization for Multi-Agent Collaboration: A Data-Driven Approach," *arXiv preprint arXiv:2310.02650*, 2023.

[109] J. A. Caley and G. A. Hollinger, "Environment Prediction from Sparse Samples for Robotic Information Gathering," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[110] J. Ott, M. J. Kochenderfer, and S. Boyd, "Approximate sequential optimization for informative path planning," *arXiv preprint arXiv:2402.08841*, 2024.

[111] J. Ott, E. Balaban, and M. Kochenderfer, "Trajectory optimization for adaptive informative path planning with multimodal sensing," *arXiv preprint arXiv:2404.18374*, 2024.

[112] M. J. Kochenderfer, T. A. Wheeler, and K. H. Wray, *Algorithms for Decision Making*. MIT Press, 2022.

[113] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel *et al.*, "Mastering chess and shogi by self-play with a general reinforcement learning algorithm," *arXiv preprint arXiv:1712.01815*, 2017.

[114] ——, "A general reinforcement learning algorithm that masters chess, shogi, and go through self-play," *Science*, vol. 362, no. 6419, pp. 1140–1144, 2018.

[115] A. Hill, A. Raffin, M. Ernestus, A. Gleave, A. Kanervisto, R. Traore, P. Dhariwal, C. Hesse, O. Klimov, A. Nichol, M. Plappert, A. Radford, J. Schulman, S. Sidor, and Y. Wu, "Stable baselines," https://github.com/hill-a/stable-baselines, 2018.

[116] DeepMind, I. Babuschkin, K. Baumli, A. Bell, S. Bhupatiraju, J. Bruce, P. Buchlovsky, D. Budden, T. Cai, A. Clark, I. Danihelka, A. Dedieu, C. Fantacci, J. Godwin, C. Jones, R. Hemsley, T. Hennigan, M. Hessel, S. Hou, S. Kapturowski, T. Keck, I. Kemaev, M. King, M. Kunesch, L. Martens, H. Merzic, V. Mikulik, T. Norman, G. Papamakarios, J. Quan, R. Ring, F. Ruiz, A. Sanchez, L. Sartran, R. Schneider, E. Sezener, S. Spencer, S. Srinivasan, M. Stanojević, W. Stokowiec, L. Wang, G. Zhou, and F. Viola, "The DeepMind JAX Ecosystem," 2020. [Online]. Available: http://github.com/deepmind

[117] M. Egorov, Z. N. Sunberg, E. Balaban, T. A. Wheeler, J. K. Gupta, and M. J. Kochenderfer, "POMDPs.jl: A framework for sequential decision making under uncertainty," *Journal of Machine Learning Research*, vol. 18, no. 26, pp. 1–5, 2017. [Online]. Available: http://jmlr.org/papers/v18/16-300.html

[118] J. Ansel, E. Yang, H. He, N. Gimelshein, A. Jain, M. Voznesensky, B. Bao, P. Bell, D. Berard, E. Burovski, G. Chauhan, A. Chourdia, W. Constable, A. Desmaison, Z. DeVito, E. Ellison, W. Feng, J. Gong, M. Gschwind, B. Hirsh, S. Huang, K. Kalambarkar, L. Kirsch, M. Lazos, M. Lezcano, Y. Liang, J. Liang, Y. Lu, C. Luk, B. Maher, Y. Pan, C. Puhrsch, M. Reso, M. Saroufim, M. Y. Siraichi, H. Suk, M. Suo, P. Tillet, E. Wang, X. Wang, W. Wen, S. Zhang, X. Zhao, K. Zhou, R. Zou, A. Mathews, G. Chanan, P. Wu, and S. Chintala, "PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation," in *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM, Apr. 2024. [Online]. Available: https://pytorch.org/assets/pytorch2-2.pdf

[119] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, R. Jozefowicz, Y. Jia, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, M. Schuster, R. Monga, S. Moore, D. Murray, C. Olah, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow, Large-scale machine learning on heterogeneous systems," Nov. 2015.

[120] H. Bayerlein, M. Theile, M. Caccamo, and D. Gesbert, "Multi-UAV Path Planning for Wireless Data Harvesting With Deep Reinforcement Learning," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 1171–1187, 2021.

[121] P. Yang, Y. Liu, S. Koga, A. Ashgharivaskasi, and N. Atanasov, "Learning Continuous Control Policies for Information-Theoretic Active Perception," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[122] L. Bartolomei, L. Teixeira, and M. Chli, "Semantic-aware active perception for UAVs using deep reinforcement learning," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021.

[123] L. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[124] H. Qie, D. Shi, T. Shen, X. Xu, Y. Li, and L. Wang, "Joint optimization of multi-UAV target assignment and path planning based on multi-agent reinforcement learning," *IEEE access*, vol. 7, pp. 146 264–146 272, 2019.

[125] T. Fan, P. Long, W. Liu, and J. Pan, "Distributed multi-robot collision avoidance via deep reinforcement learning for navigation in complex scenarios," *Intl. Journal of Robotics Research (IJRR)*, vol. 39, no. 7, pp. 856–892, 2020.

[126] A. Arora, P. M. Furlong, R. Fitch, S. Sukkarieh, and T. Fong, "Multi-modal active perception for information gathering in science missions," *Autonomous Robots*, vol. 43, pp. 1827–1853, 2019.

[127] B. Bucher, K. Schmeckpeper, N. Matni, and K. Daniilidis, "An Adversarial Objective for Scalable Exploration," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021, pp. 2670–2677.

[128] J. Ott, S.-K. Kim, A. Bouman, O. Peltzer, M. Sobue, H. Delecki, M. J. Kochenderfer, J. Burdick, and A.-a. Agha-mohammadi, "Risk-aware meta-level decision making for exploration under uncertainty," *arXiv preprint arXiv:2209.05580*, 2022.

[129] A. Bouman, J. Ott, S.-K. Kim, K. Chen, M. J. Kochenderfer, B. Lopez, A.-a. Agha-mohammadi, and J. Burdick, "Adaptive coverage path planning for efficient exploration of unknown environments," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 11 916–11 923.

[130] X. Ye, Z. Lin, H. Li, S. Zheng, and Y. Yang, "Active Object Perceiver: Recognition-guided Policy Learning for Object Searching on Mobile Robots," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2018.

[131] D. A. Pomerleau, "Efficient training of artificial neural networks for autonomous navigation," *Neural Computation*, vol. 3, no. 1, pp. 88–97, 1991.

[132] B. Settles, "Active learning literature survey," University of Wisconsin-Madison, Department of Computer Sciences, Tech. Rep., 2009.

[133] J. Kiefer, "Optimum experimental designs," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 21, no. 2, pp. 272–304, 1959.

[134] J. Rückin, L. Jin, F. Magistri, C. Stachniss, and M. Popović, "Informative Path Planning for Active Learning in Aerial Semantic Mapping," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.

[135] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya *et al.*, "A review of uncertainty quantification in deep learning: Techniques, applications and challenges," *Information fusion*, vol. 76, pp. 243–297, 2021.

[136] L. Zhu, J. Cheng, H. Zhang, Z. Cui, W. Zhang, and Y. Liu, "Multi-robot Collaborative Area Search Based on Deep Reinforcement Learning," *arXiv preprint arXiv:2312.01747*, 2023.

[137] R. Menon, T. Zaenker, N. Dengler, and M. Bennewitz, "NBV-SC: Next Best View Planning based on Shape Completion for Fruit Mapping and Reconstruction," in *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[138] C. Cao, H. Zhu, F. Yang, Y. Xia, H. Choset, J. Oh, and J. Zhang, "Autonomous Exploration Development Environment and the Planning Algorithms," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[139] T. Novkovic, R. Pautrat, F. Furrer, M. Breyer, R. Siegwart, and J. Nieto, "Object Finding in Cluttered Scenes Using Interactive Perception," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[140] K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige *et al.*, "Using simulation and domain adaptation to improve efficiency of deep robotic grasping," in *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2018.

[141] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: a survey," in *2020 IEEE symposium series on computational intelligence (SSCI)*, 2020.

[142] H. Zhu, J. Yu, A. Gupta, D. Shah, K. Hartikainen, A. Singh, V. Kumar, and S. Levine, "The ingredients of real-world robotic reinforcement learning," in *Proc. of the Int. Conf. on Learning Representations (ICLR)*, 2020.

[143] R. Kirk, A. Zhang, E. Grefenstette, and T. Rocktäschel, "A survey of zero-shot generalisation in deep reinforcement learning," *Journal of Artificial Intelligence Research (JAIR)*, vol. 76, pp. 201–264, 2023.

[144] A. Gupta, R. Mendonca, Y. Liu, P. Abbeel, and S. Levine, "Meta-reinforcement learning of structured exploration strategies," in *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, 2018.

[145] F. Sadeghi and S. Levine, "CAD2RL: Real Single-Image Flight Without a Single Real Image," in *Proc. of Robotics: Science and Systems (RSS)*, 2017.

[146] S. Genc, S. Mallya, S. Bodapati, T. Sun, and Y. Tao, "Zero-Shot Reinforcement Learning with Deep Attention Convolutional Neural Networks," in *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, 2020.