A Survey on Active Simultaneous Localization and Mapping: State of the Art and New Frontiers

Julio A. Placed , *Student Member, IEEE*, Jared Strader , Henry Carrillo, Nikolay Atanasov , *Member, IEEE*,
Vadim Indelman , *Member, IEEE*, Luca Carlone , *Senior Member, IEEE*,
and José A. Castellanos , *Senior Member, IEEE*

Abstract—Active simultaneous localization and mapping (SLAM) is the problem of planning and controlling the motion of a robot to build the most accurate and complete model of the surrounding environment. Since the first foundational work in active perception appeared, more than three decades ago, this field has received increasing attention across different scientific communities. This has brought about many different approaches and formulations, and makes a review of the current trends necessary and extremely valuable for both new and experienced researchers. In this article, we survey the state of the art in active SLAM and take an in-depth look at the open challenges that still require attention to meet the needs of modern applications. After providing a historical perspective, we present a unified problem formulation and review the well-established modular solution scheme, which decouples the problem into three stages that identify, select, and execute potential navigation actions. We then analyze alternative approaches, including belief-space planning and deep reinforcement learning techniques, and review related work on multirobot coordination. This article concludes with a discussion of new research directions, addressing reproducible research, active spatial perception, and practical applications, among other topics.

Index Terms—Active perception, active simultaneous localization and mapping (SLAM), autonomous robotic exploration, belief-space planning (BSP), deep reinforcement learning (DRL), next best view, optimality criteria.

I. INTRODUCTION

AUTONOMOUS operation in robotics applications requires robots to have access to a consistent model of the surrounding environment, in order to support safe planning and decision making. Toward this goal, a robot must have the ability to create a map of the environment, localize itself on it, and control its own motion. Active simultaneous localization and mapping (SLAM) refers to the joint resolution of these three core problems in mobile robotics, with the ultimate goal of creating the most accurate and complete model of an unknown environment. Active SLAM can be seen as a decision-making process in which the robot has to choose its own future control actions, balancing between exploring new areas and exploiting those already seen to improve the accuracy of the resulting map model.

During the last decades, active SLAM has received increasing attention[1] and has been studied in different forms across multiple communities, with the ambition of deploying autonomous agents in real-world applications (e.g., search and rescue in hazardous environments, underground, or planetary exploration). This divergence has broadened the scope of the problem and provided a wider context, yielding numerous approaches based on different concepts and theories that have made the field flourish; but it also created a disconnect between research lines that could mutually benefit from each other. With this survey, we seek to fill this gap by providing a general problem statement and a unified review of related works.

Currently, active SLAM is at a decisive point, driven by novel opportunities in spatial perception and artificial intelligence (AI). These include, for instance, the application of breakthroughs in neural networks to prediction beyond line-of-sight, reasoning over novel environment representations, or leveraging new SLAM techniques to process dynamic and deformable scenes. Throughout this article, we give a fresher picture of active SLAM that goes beyond the classical—but still mainstream—entropy computation over discretized grids. Besides, we identify the open challenges that need to be addressed for active SLAM to have an impact on real applications, shaping future lines of research, and describing how they can nourish from the cross-fertilization between research fields. Among those challenges, we emphasize the urgent need for benchmarks and reproducible research.

---

[1]The number of publications on active SLAM has grown from 53 in 2010 to over 660 in 2022 (a 12-fold increase). The number becomes almost 5500 if we extend the search to include BSP, active exploration, and simultaneous planning, localization, and mapping. Source: dimensions.ai.

A. Historical Perspective

Ever since the first mobile robots were built in the late 1940s, the ambition that they could perform autonomous tasks has been one of the major focuses of robotics research. To operate autonomously, a robot needs to form a model of the surrounding environment—including localization and mapping—and perform safe navigation [1]. While the former involves estimating the position of the robot and creating a symbolic representation of the environment, the latter refers to planning and controlling the movements of the robot to safely achieve a goal location. Localization, mapping, and planning have been often investigated in combination, resulting in multiple research areas, such as SLAM, active localization, active mapping, and active SLAM.

Localization and mapping were treated deterministically and solved independently until probabilistic approaches went mainstream in the 1990s, when researchers realized that both tasks were correlated and dependent of one another. SLAM refers, thereby, to the problem of incrementally building the map of an environment while at the same time locating the robot within it [2]. This problem has attracted significant attention from the robotics community in the last decades; see [3], [4], [5] and the references therein.

SLAM, however, is a passive method and is not concerned with guiding the navigation process. In contrast, *active* approaches do consider the navigation aspects of the problem. Bajcsy [6], Cowan and Kovesi [7], and Aloimonos et al. [8] were the first to study and analyze the problem of active perception (also referred to as active information acquisition [9]) in the late 1990s. Bajcsy [10] would later formally define it as the problem of actively acquiring data in order to achieve a certain goal, necessarily involving a decision-making process. For the cases in which the objective is to improve localization, mapping, or both, the problems are, respectively, referred to as active localization, active mapping, and active SLAM.

Active mapping was the first problem to be addressed, dating back to the work of Connolly [11] in 1985. Better known since then as the *next best view* problem, active mapping tackles the search of the optimal movements to create the best possible representation of an environment. Subsequent examples date to the 1990s [12], [13], [14], always under the assumption of perfectly known sensor localization. This problem has been primarily addressed in the computer vision community to reconstruct objects and scenes from multiple viewpoints, since the nature of the projective geometry for monocular cameras, occlusions, and limited field of view often make impossible to do it from just one viewpoint; see [15] and the references therein.

In a similar vein, *active localization* aims to improve the estimation of the robot's pose by determining how it should move, assuming the map of the environment is known. First relevant works can be traced back to 1998, when Fox et al. [16] and Borgi and Caglioti [17] formulated it as the problem of determining the robot motion so as to minimize its future expected (i.e., a posteriori) uncertainty. In particular, it is in [16] where the foundations of the current workflow were laid:
1) goal identification;
2) utility computation;
3) action selection (we will extensively review these stages later in this survey).

Other relevant subsequent work can be found in [18], [19], [20], and [21], but also in the related literature of perception-aware planning [22] and planning under uncertainty [23].

Finally, *active SLAM* unifies the previous problems, and allows a robot to operate autonomously in an initially unknown environment. It refers to the application of active perception to SLAM and can be defined as the problem of controlling a robot, which is performing SLAM in order to reduce the uncertainty of its localization and the map representation [24]. Historically, active SLAM has been referred to with different terminology, which has significantly hindered knowledge sharing and dissemination within the robotics community. Relevant seminal works can be found under the names of active exploration [25], adaptive exploration [26], [27], integrated exploration [28], [29], autonomous SLAM [30], simultaneous planning, localization and mapping [31], belief-space planning (BSP) [32], or simply robotic exploration [33], [34]. It was not until 2002—when Davison and Murray [35] coined the term active SLAM—that the robotics community started adopting this nomenclature. Thrun and Möller [25] demonstrate that in order to solve robotic exploration, agents have to switch between two opposite principles depending on the expected costs and gains: exploring new areas and revisiting those already seen, i.e., the so-called *exploration–exploitation dilemma*. The first approach in which a robot chooses actions that maximize the knowledge of the two variables of interest is attributed to Feder et al. [26], who also separate the procedure in three major stages as in [16]. Table I contains a subset of relevant works that have followed [26]. This table differentiates the main aspects of each approach, including the type of sensors, the state representation, and the theoretical foundations.

B. About Previous Surveys

Only two works have previously addressed the problem of surveying active SLAM research. The first of them, published in 2016, is a section of a more general survey on SLAM carried out by Cadena et al. [5]. The other, by Lluvia et al. [36], conducts a more extensive survey on "active mapping and robot exploration." Table II summarizes the topics they address, along with those covered in the present survey.

Cadena et al. [5] described both the history and the main aspects of the problem, and identify three open challenges: the decision of when to stop performing active SLAM, the problem of accurately predicting the effect of future actions, and the lack of mathematical guarantees of optimality. However, the brevity of the active SLAM section prevented delving into a detailed discussion of the most relevant works or providing a more unified mathematical formulation of the problem. Moreover, since the work [5] was published, many relevant contributions have been proposed and new open problems have arisen. For instance, progress has been made on the way uncertainties of the robot location and the map are represented and quantified. Furthermore, recent work has also opened new research endeavors, including deep learning (DL).

Lluvia et al. [36] also provided a thorough historical review and relate the different communities that have been trying to solve this problem under different nomenclatures. Similar to [5], they do not attempt to present a unified mathematical formulation of active SLAM nor do they cover utility computation, a field which has been mostly overlooked in the literature. They delve, nevertheless, into the optimization of vantage points and the trajectories to reach them, a new problem that has attracted significant attention from the control community and has seen

TABLE I
COMPARISON BETWEEN REPRESENTATIVE ACTIVE SLAM APPROACHES, ORDERED CHRONOLOGICALLY

Reference	SLAM Approach	Sensors	Environment Representation	Formulation	Candidate Goal Locations	Utility Function	Validation Environment	Stopping Criterion	Publicly available
Feder et al. [26]	EKF	Sonar	Landmark map	Modular	Local vicinity	D-opt	Sim. and real	-	✗
Bourgault et al. [27]	EKF	Lidar	OG map	Modular	Local vicinity	MI	Real	-	✗
Stachniss et al. [29]	FastSLAM [37]	Lidar	OG map	Modular	Frontiers and revisiting	Particle's volume and distance	Sim. and real	Particle's volume	✗
Stachniss et al. [38]	RBPF	Lidar	OG map	Modular	Frontiers and revisiting	MI and distance	Sim. and real	-	✗
Leung et al. [39]	EKF	Lidar	Landmark map	MPC	Unknown space and revisiting	T-opt	Simulation	-	✗
Valencia et al. [40]	Pose SLAM [41]	Lidar	OG map	Modular	Frontiers and revisiting	Entropy	Simulation	-	✗
Carlone et al. [42]	RBPF	Lidar	OG map	Modular	Frontiers and revisiting	KLD	Simulation	-	✗
Indelman et al. [43]	GTSAM	Camera	Landmark map	BSP	-	Robot's T-opt and distance	Simulation	-	✗
Zhu et al. [44]	RGBDSLAM [45]	RGB-D	Octomap	Modular	Frontiers	Coverage and distance	Simulation	-	✗
Bircher et al. [46]	ROVIO [47] and mapping	Stereo and IMU	Octomap	MPC	RRT paths	Coverage and distance	Sim. and real	Coverage	✓
Papachristos et al. [48]	ROVIO [47] and dense mapping	Stereo and IMU	Octomap	MPC	RRT paths	D-opt	Sim. and real	Min. utility	✓
Umari et al. [49]	Gmapping [50]	Lidar	OG map	Modular	Frontiers	Map's MI and distance	Sim. and real	-	✓
Carrillo et al. [51]	ICP and iSAM [52]	Lidar	OG map	Modular	Frontiers	Shannon-Rényi entropy	Sim. and real	Time	✓
Jadidi et al. [53]	Pose SLAM [41]	Lidar	COM	Modular	Frontiers	MI and distance	Simulation	Coverage	✓
Palomeras et al. [54]	ICP and g2o [55]	Lidar	Octomap	Modular	Random	Coverage	Sim. and real	Coverage	✗
Chaplot et al. [56]	Neural Networks	RGB	OG map	DRL	Local vicinity	Coverage	Sim. and real	-	✓
Niroui et al. [57]	Gmapping [50]	Lidar	OG map	DRL	Frontiers	Map's MI and distance	Sim. and real	-	✗
Chen et al. [58]	GTSAM	Range	Virtual landmark map	DRL	Frontiers	Virtual landmark's T-opt	Simulation	-	✓
Li et al. [59]	Karto [60] and g2o [55]	Lidar	OG map	DRL	Sampled from the OG map	Map's MI and distance	Sim. and real	Coverage	✗
Suresh et al. [61]	ICP and iSAM [52]	Sonar	Octomap	MPC	RRT paths and revisiting	Robot's D-opt and coverage	Sim. and real	-	✗
Chen et al. [62]	Linear SLAM [63]	Camera	Landmark map	MPC	Local vicinity	Graph's D-opt	Sim. and real	Coverage	✗
Batinovic et al. [64]	Cartographer [65]	Lidar and IMU	Octomap	Modular	Frontiers	Map's MI and distance	Sim. and real	Coverage	✓
Placed et al. [66]	ORB-SLAM2 [67]	Lidar and RGB-D	Octomap	Modular	Frontiers	Graph's D-opt	Simulation	Time	✓
Bonetto et al. [68]	RTAB-Map [69]	Lidar, RGB-D and IMU	Octomap	Modular and MPC	Frontiers	Map's MI, distance and visual features	Sim. and real	Time	✓

TABLE II
COMPARISON BETWEEN THE TOPICS AND OPEN CHALLENGES ADDRESSED IN
PREVIOUS SURVEYS AND THE CURRENT ONE

	Topic	Cadena et al. [5]	Lluvia et al. [36]	Ours
Introduction	Historical review	Briefly	Yes	Yes
	Problem formulation	No	No	Yes
Modular scheme	Env. representation	Yes	Yes	Yes
	Goal identification	Briefly	Yes	Yes
	IT	Briefly	No	Yes
	TOED	Briefly	No	Yes
	Graph Theory	No	No	Yes
Alternative approaches	Continuous domain	No	Yes	Yes
	DL	No	Briefly	Yes
	Multirobot	No	No	Yes
Open problems	State prediction	Yes	Yes	Yes
	Stopping criteria	Yes	Briefly	Yes
	Novel representations	No	Briefly	Yes
	Data association	No	No	Yes
	Complex environments	No	No	Yes
	Reproducible research	No	No	Yes
	Practical applications	No	No	Yes

many contributions in recent years. In [36], the authors present a comparison between representative works in active SLAM, although with a limited scope. Contrarily to [36], we present a more complete analysis and a broader set of open challenges, which extends the ones identified in [5].

C. Article Structure

The rest of this article is organized as follows. Section II provides a unified problem formulation for active SLAM and describes the three subproblems (or *stages*) it has traditionally been divided into. Sections III–V cover those three stages separately. In particular, Section III deals with the identification of vantage points, Section IV with utility computation, and Section V with selection and execution of the optimal action. Sections VI and VII consider, on the other hand, alternative continuous-state optimization and DL methods. Section VIII is devoted to multirobot active SLAM. Section IX outlines the open research questions in active SLAM. Finally, Section X concludes this article.

II. ACTIVE SLAM PROBLEM

A. Problem Formulation

Active SLAM can be framed within the wider mathematical framework of partially observable Markov decision processes (POMDPs), after some particularization. POMDPs model decision-making problems under both action and observation uncertainties and can be formally defined as the 7-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{Z}, \xi_s, \xi_z, r, \gamma)$. In particular, a POMDP consists of the agent's state space $\mathcal{S}$, a set of actions $\mathcal{A}$, a transition function between states $\xi_s : \mathcal{S} \times \mathcal{A} \mapsto \Pi(\mathcal{S})$ where $\Pi(\mathcal{S})$ is the space of probability density functions (pdfs) over $\mathcal{S}$, an observation space $\mathcal{Z}$, the conditional likelihood of making any of those observations $\xi_z : \mathcal{S} \mapsto \Pi(\mathcal{Z})$, where $\Pi(\mathcal{Z})$ is the space of pdfs over $\mathcal{Z}$, a

reward scalar mapping $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, and the discount factor $\gamma \in (0, 1) \in \mathbb{R}$, which allows us to work with finite rewards even when planning over infinite time horizons.

Contrary to the fully observable case, agents in a POMDP cannot reliably determine their own true state, $\boldsymbol{s}$. Instead, they maintain an internal *belief* or *information state*, $b_t(\boldsymbol{s}_t)$, which represents the posterior probability over states at time t, given the available data collected up to that time [2], [70], [71]

$$b_t(\boldsymbol{s}_t) \triangleq p(\boldsymbol{s}_t | \underbrace{\boldsymbol{z}_{1:t}, \boldsymbol{a}_{1:t-1}}_{\text{history, } \boldsymbol{h}}) \qquad (1)$$

where $\boldsymbol{z}_{1:t}$ is the set of all available observations and $\boldsymbol{a}_{1:t-1}$ the set of past control actions (both collectively referred as the *history* $\boldsymbol{h}$). The *belief space*, $\mathcal{B}(\mathcal{S}) \equiv \Pi(\mathcal{S})$, of pdfs over the set $\mathcal{S}$ is defined as

$$\mathcal{B}(\mathcal{S}) \triangleq \left\{ b : \mathcal{S} \mapsto \mathbb{R} \ \middle| \ \int b(s) ds = 1, \ b(s) \geq 0 \right\}. \qquad (2)$$

In order to evaluate the effect of future actions, agents must be capable of predicting posterior belief distributions, that is, the pdf over $\mathcal{S}$ after performing a certain action, $\boldsymbol{a}_t$, and taking a future observation $\boldsymbol{z}_{t+1}$

$$b_{t+1}(\boldsymbol{s}_{t+1}) \triangleq p(\boldsymbol{s}_{t+1} | \boldsymbol{z}_{t+1}, \boldsymbol{a}_t, b_t(\boldsymbol{s}_t)). \qquad (3)$$

Since the future measurements are unknown for the agent, their expected value has to be studied instead. Consider that an agent in the state defined by $b_t(\boldsymbol{s}_t)$ executes a certain action $\boldsymbol{a}_t$, and transitions to another state with pdf $p(\boldsymbol{s}_{t+1})$. Then, the likelihood of making an observation will be given by [71]

$$p(\boldsymbol{z}_{t+1} | b_t(\boldsymbol{s}_t), \boldsymbol{a}_t) = \int \int \xi_z(\boldsymbol{s}_{t+1}) \, \xi_s(\boldsymbol{s}_t, \boldsymbol{a}_t) \\ b_t(\boldsymbol{s}_t) \, d\boldsymbol{s}_t \, d\boldsymbol{s}_{t+1} \qquad (4)$$

where $\xi_z(\boldsymbol{s}_{t+1}) = p(\boldsymbol{z}_{t+1} | \boldsymbol{s}_{t+1})$ is the observation model and $\xi_s(\boldsymbol{s}_t, \boldsymbol{a}_t) = p(\boldsymbol{s}_{t+1} | \boldsymbol{s}_t, \boldsymbol{a}_t)$ the motion model.

Since the belief is a sufficient statistic, optimal policies for the original POMDP may be found by solving an equivalent continuous-space MDP over $\mathcal{B}(\mathcal{S})$ [70], [72]. Such MDP is defined by the 5-tuple $(\mathcal{B}, \mathcal{A}, \xi_b, \rho, \gamma)$, where the transition and reward functions are $\xi_b : \mathcal{B} \times \mathcal{A} \mapsto \Pi(\mathcal{B})$ and $\rho : \mathcal{B} \times \mathcal{A} \mapsto \mathbb{R}$. To preserve consistency, this belief-dependent reward function builds on the expected rewards of the original POMDP

$$\rho(b_t, \boldsymbol{a}_t) = \int_{\mathcal{S}} b_t(\boldsymbol{s}_t) \, r(\boldsymbol{s}_t, \boldsymbol{a}_t) \, d\boldsymbol{s}_t. \qquad (5)$$

Then, the decision at time t will be provided by the (control/action) policy π_t, which maps elements from the space of pdfs over $\mathcal{S}$ to the action space

$$\pi_t : \mathcal{B}(\mathcal{S}) \mapsto \mathcal{A}. \qquad (6)$$

The optimal policy, $\pi^\star$ that yields the highest expected rewards for every belief state can be found via

$$\pi^\star(b) = \arg\max_\pi \sum_{t=0}^\infty \mathbb{E}\left[\gamma^t \rho(b_t, \pi(b_t)) \right] \qquad (7)$$

where expectation is taken w.r.t. $p(\boldsymbol{z}_{t+1} | b_t(\boldsymbol{s}_t), \boldsymbol{a}_t)$. In general, computing the optimal policy for MDPs with continuous state spaces is hard and most works resort to approximate solutions or problem simplifications [70], [73].

The active SLAM problem requires, however, some variation and particularization of the above general POMDP formulation. Let us consider a robot capable of moving in an unknown environment while performing SLAM. That is, at every time step, the robot can change its own linear and angular velocities; moreover, the robot is able to process the sensor data into a map representation, $\boldsymbol{m}_t \in \mathcal{M}$, and an estimate of its own state (e.g., pose), $\boldsymbol{x}_t \in \mathcal{X}$. Thus, the state space can be defined as the joint space $\mathcal{S} \triangleq \mathcal{X} \times \mathcal{M}$.

The evolution of both the state and the measurements in SLAM is governed by probabilistic laws [2], as (1) and (4) express. However, two assumptions are worth mentioning in the context of active SLAM regarding each of the equations that further simplify its resolution. First, the robot state is commonly assumed Gaussian with a pdf $b(\boldsymbol{x})$ having mean $\hat{\boldsymbol{x}}$ and covariance $\boldsymbol{\Sigma}_r$ (see, e.g., [40], [43]). Thus, the map and the robot state are usually treated independently, although some representations allow for a joint distribution (e.g., in sparse landmark maps or using Gaussian Processes to model dense maps [53]). Second, despite less prevalent than the former, some works (e.g., [32]) also assume *maximum likelihood* (ML) observations, i.e., that executing an action in a given belief state will always produce the same, most probable observation. This allows us to rewrite the expected measurements as

$$\boldsymbol{z}_{t+1}^{\text{ML}} = \arg\max_{\boldsymbol{z} \in \mathcal{Z}} \ p(\boldsymbol{z}_{t+1} | b_t(\boldsymbol{s}_t), \boldsymbol{a}_t). \qquad (8)$$

In addition, in active SLAM the reward typically reflects the agent's knowledge of the system (i.e., it involves the uncertainty in the belief rather than focusing on reaching specific states). These reward functions are known as *utility functions* and may be defined mathematically as the scalar mapping $\rho : \mathcal{B}(\mathcal{S}) \times \mathcal{A} \mapsto \mathbb{R}$. This reward mapping, however, is inconsistent with both POMDPs (where the reward is dependent on $\boldsymbol{s}$ and $\boldsymbol{a}$) and belief MDPs [where the reward is restricted to the form in (5)]. To circumvent this limitation, ρ-POMDP [73] extends the POMDP formulation to allow the inclusion of beliefs' uncertainty in the objective. This enables the use of information-oriented criteria rather than control-oriented, without losing basic properties, such as Markovianity.

Finally, considering a finite-horizon and ML observations, the discount factor and expectation over future measurements in (7) can be dropped, and active SLAM can be reduced to the following optimization for open-loop planning settings:

$$\boldsymbol{a}_{t:t+k}^\star = \arg\max_{\boldsymbol{a}_{t:t+k} \in \mathcal{A}^k} \sum_{\tau=t}^{t+k} \rho\left(b(\boldsymbol{s}_\tau), \boldsymbol{a}_\tau \right) \qquad (9)$$

where $\boldsymbol{a}_{t:t+k}^\star$ is the optimal sequence of actions to execute over the future planning horizon (k lookahead steps) and $\mathcal{A}^k \triangleq \mathcal{A} \times \mathcal{A} \times \cdots \times \mathcal{A}$ the space of sequences of actions over k.

B. Decoupling Active SLAM Into Three Subproblems

While the previous section provided a unified formulation for active SLAM, for computational convenience active SLAM has been traditionally decoupled into three subproblems (or *stages*) [16], [26], [28], which will be briefly described hereafter and covered in detail in Sections III–V.

1) *Identification of the potential actions:* Solely to reduce the computational burden, the first stage aims to determine a reduced subset of possible actions to execute.

2) *Utility computation:* The expected cost and gain of performing each candidate action has to be estimated.

3) *Action selection and execution:* Finally, the last stage involves finding and executing the optimal action(s).

The entire process should be iteratively repeated until the whole environment is accurately modeled, although in practice it is done until some stopping conditions are met.

For clarity of presentation and because many existing works do decompose active SLAM into these stages, we review each stage separately in Sections III–V. However, this decoupling can produce suboptimal results and lead to undesired behaviors. Performing the three stages simultaneously is certainly advantageous, e.g., when optimizing over a continuous action space, or when a control policy is optimized or learned under the umbrella of POMDPs. We review these approaches, alternative to the modular scheme, in Sections VI and VII.

III. STAGE 1: IDENTIFICATION OF POTENTIAL ACTIONS

The first stage in modular active SLAM approaches consists in generating the set of available actions the robot could execute (i.e., goals the robot can reach); this can be understood as a way to reduce (and discretize) the search space of potential actions. Early works simply used random goals or required human interaction, until the concept of *frontiers* was introduced by Yamauchi [74]. This resulted in improved exploration strategies, and has consolidated as the most common approach. Nevertheless, the advent of neural networks has led to new ways of evaluating the space of potential goals. In this section, we present the most important methods to identify goal locations. Since they strongly depend on the representation of the environment estimated by the SLAM pipeline, we start by providing a brief description of the different existing representations for active SLAM.

A. Representation of the Environment

We review four different types of map representations: topological, metric, metric-semantic, and hybrid maps.

1) Topological Maps: Use lightweight graphs to describe information about the topology of the environment. Historically, vertices in this graph represent convex regions in the free space, while edges model connections between them. The construction of these graphs is a segmentation problem, usually done over an occupancy grid; see [75] for a survey on these methods. Despite these maps allow leveraging graph theory, which provides powerful tools for planning and exploration, they are not frequently used in active SLAM [76], [77].

2) Metric Maps: Are the most used representations to encode information about the environment in active SLAM. They can be further divided into the following two categories: sparse and dense maps. The former rely on a sparse set of interest points (or *landmarks*) to represent a scene, and have been especially used in optimal control [39], [62], [78], and BSP [43] approaches. Dense maps can be based on point clouds, meshes or, more typically, a discretization of the environment into cells that encode a certain metric (e.g., occupancy, distance to obstacles). Occupancy grid (OG) maps, first proposed in the late eighties for perception and navigation by Elfes [79] and Moravec [80], assign to each cell its probability of being occupied. They have been used in numerous active SLAM frameworks, e.g., [51], [56], [57], [81]. Their extension to 3-D include OctoMaps [82],

Supereight [83], and voxel maps [84], all of which have been also used in active SLAM [85], [86], [87], [88]. Jadidi et al. [53] used continuous occupancy maps (COM) to leverage continuous optimization methods. There exist many other dense maps that encode more sophisticated metrics, such as those based on signed distance fields (SDF), such as Voxblox [89]. Still, they are seldom used in active SLAM [90].

3) Metric-Semantic Maps: Go beyond geometric modeling and associate semantic information to classical metric maps. Instead of geometric features, a sparse map can capture objects, described by a semantic category, pose, and shape [91], [92]. Active object-level SLAM has been considered in [93], [94]. Examples of dense metric-semantic maps include Voxblox++ [95] and Kimera [96] (which build upon an SDF), and Fusion++ [97] and [98] (based on voxel maps). Despite being used in some SLAM formulations (see [5], [96] and the references therein), they have not yet been used in active SLAM. An exception is the work of Asgharivaskasi and Atanasov [99], [100], which develops a multiclass (semantic) OctoMap and uses a closed-form lower bound on the Shannon mutual information (MI) between the map and range-category observations to select informative robot trajectories.

4) Hybrid and Hierarchical Maps: Combine some of the previous representations to enhance the decision-making process. Hybrid metric-topological maps have been applied to tackle either navigation [101] or SLAM [102]. Rosinol et al. [103] combined metric, semantic, and topological representations into a single model, a *3-D scene graph*. These hierarchical representations break down metric-semantic maps into interconnected high-level entities, paving the way for high-level reasoning. The use of hybrid maps in active SLAM is mostly unexplored, with [104] among the few works that have integrated them.

B. Detecting Goal Locations

The identification of all possible destinations the robot could travel to easily proves to be intractable because of the dimensions of the map and the action set [105]. In practice, a finite subset of them is identified, allowing for computational tractability despite not guaranteeing global optimality [43]. The simplest approach consists of randomly selecting the goal destinations [106], [107]. *Random exploration* requires low computational resources and works under the assumption that every spot in the environment has the same information associated. In 1997, Yamauchi [74] revolutionized the field by introducing the concept of *frontiers*, i.e., the areas that lie between known and unknown regions. Since its proposal, *frontier-based exploration* has been the most used by far and has been tailored to different map representations. Frontiers have been effectively identified for topological maps as nodes with no neighbors in certain directions [76]. For 2-D OG maps, a plethora of geometric frontier-detection methods have been developed to circumvent the computational cost of searching the entire space [108]. Keidar and Kaminka [109] proposed the wavefront frontier detector (WFD) and fast frontier detector (FFD). WFD starts the search from the robot's location and restricts it to the free space; FFD performs the search after each scan is collected, following the intuition that frontiers are bound to appear in recently scanned regions. Following this idea, the same authors present the incremental WFD [110] that restricts the search to recently scanned areas. Quin et al. [108] improved the performance of the previous algorithms by only evaluating a subset of the observed

free space. Refer to [108], [111] for further discussion. Umari and Mukhopadhyay [49] first present a frontier search method over a 2-D OG based on rapidly-exploring random trees (RRTs) that grow both globally and locally to sample recently scanned regions. This strategy, often combined with computer vision algorithms has been widely used [81], [112]. The sample-based frontier detector algorithm [113] reduces the computational load of the previous methods by only storing the nodes of the search tree. Frontier identification in 3-D maps is less frequent, since 3-D maps are more expensive to store and analyze, and are often incomplete due to the sensed volume. Apart from simple search techniques [88], [114], most methods evaluate map portions incrementally [44], [64] or along surfaces [115]. Alternatively, in [116], authors proposed a method that disperse random particles over the 3-D known space. No matter the method used, after detecting frontiers, a clustering step is frequently required to prevent the frontier set from being high-dimensional (e.g., using K-means [117] or mean-shift [49]).

Shortly after the concept of frontiers was proposed, Newman et al. [30] and Stachniss et al. [29] realized that, for a robot with high uncertainty, *potential loop closure* areas encode more information than frontiers; the ultimate goal of active SLAM goes beyond simply covering the workspace: to improve the accuracy of localization and mapping. Similarly, Grabowski et al. [118] observed that regions of interest where sensor readings overlap may be more informative than new frontiers. In other words, these works explicitly account for the *exploration–exploitation dilemma* in the frontier detection step. It is a common practice in active SLAM to include potential loop closure regions—along with frontiers—in the set of goal candidates [40], [54], or to switch between exploring new frontiers and revisiting known places [29], [61], [119].

In contrast to frontier-based approaches, some active SLAM formulations allow the identification of goal locations locally in the *robot's vicinity*. However, note that decisions will be optimal only locally and a short decision-making horizon may induce wrong behaviors [26], [120]. This strategy is typical in deep reinforcement learning (DRL) approaches [121], [122], [123], for which local optimality is alleviated by network memorization. Following the idea that evaluating larger neighborhoods would lead to more robust decisions, in [40] authors use RRT-based paths to several configurations over the free space as the action set; and in [43], the entire environment is considered under the umbrella of continuous-domain optimization.

IV. STAGE 2: UTILITY COMPUTATION

The second and main stage in modular active SLAM approaches focuses on the evaluation of each possible destination, in order to estimate the effect that executing the set of actions to reach each destination would have. Naive utility formulations using just geometric or time-dependent functions often result in nondesirable behaviors [40], [81], [124], since they do not properly capture the uncertainty in the belief. The *exploration–exploitation dilemma* can be more effectively solved by quantifying the expected uncertainty of the two target random variables: the robot location and the map. Typically, the different objectives (e.g., travelling cost, mapping, and localization uncertainty) are aggregated into a single utility function, although there are multiobjective approaches in which they are kept separate and Pareto optimal solutions are sought [125], [126], [127]. There is a plethora of metrics and the choice of which one to use mainly

depends on the selected way to represent the variables of interest. Metrics based on information theory (IT) usually aim at OG maps, while those based on the theory of optimal design are more suitable for Gaussian distributions. We review each choice as follows.

A. Naive Cost Functions

The simplest (and first-broadly-used) metrics are naive geometric functions, such as the Euclidean distance to the goal location [74], the time required to reach it [88], or the expected size of the are to visit [49], [106], [107]. In fact, the latter approximates the map's entropy, which is strongly related to the number of known cells in an OG map [28]. Since these metrics are computed over Euclidean or temporal spaces, they can be used regardless of the map representation chosen [54], [64], [104]. Stachniss et al. [38] showed that combining distance and information-based functions results in better exploration strategies, and this has since been a common approach [128]. However, manual tuning to overcome discrepancies between the multiple terms involved is needed [49], [129].

B. Information Theory

The most common approach to assess utility in active SLAM uses IT to quantify the uncertainty in the joint belief state. Within it, there exist different metrics that allow for such quantification, although all of them build on the same concept: *entropy*. The notion of entropy was introduced by Shannon [130] and can be defined as a measure of a variable's uncertainty, randomness, or surprise; this is in fact strongly related to its associated information [131].

Early exploration strategies use only the map representation as the variable of interest [74], [88], [132], thereby assuming no error in the robot localization. However, soon after the first of these works emerged, it was observed that high uncertainty in the robot state estimation leads to wrong expected map uncertainties [27]. The entropy of the SLAM posterior after executing a candidate action can be computed as [38]

$$\mathcal{H}\left[p(\boldsymbol{x}, \boldsymbol{m} | \boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a})\right] \triangleq$$

$$\underbrace{\mathcal{H}\left[p(\boldsymbol{x} | \boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a})\right]}_{\text{robot's } \mathcal{H}} + \underbrace{\int_{\boldsymbol{x}} p(\boldsymbol{x} | \boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a}) \mathcal{H}\left[p(\boldsymbol{m} | \boldsymbol{x}, \boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a})\right] d\boldsymbol{x}}_{\text{expected conditional map's } \mathcal{H}}$$

$$(10)$$

where $\hat{\boldsymbol{z}}$ are the expected (ML) future measurements, which may be estimated using, e.g., ray-casting techniques [24].

The computation of the previous joint entropy is intractable in general [38]. To overcome this, most approaches resort to entropy approximations that first compute utility of the two variables independently, and then combine them heuristically [27], [38], [61], [133]. Let us first consider the case of graph-based SLAM, in which the problem is described using a graph representation where nodes represent the robot poses and edges encode the constraints between them; see [2], [3], [4], [5]. The joint entropy in (10) can be approximated by [40]

$$\mathcal{H}\left[p(\boldsymbol{x}, \boldsymbol{m} | \boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a})\right] \approx \mathcal{H}\left[p(\boldsymbol{x} | \boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a})\right] + \mathcal{H}\left[p(\boldsymbol{m} | \boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a})\right].$$

$$(11)$$

The mismatch between the magnitudes of the addends above is the main drawback of such approximation, calling for the

addition of weighting parameters to balance the contributions of the two terms [42], [134]. Carrillo et al. [51] circumvent this by embedding a metric of the robot's uncertainty in a combined Shannon-Rényi utility function; an approach that also appears in [135]. On the other hand, the expectation–maximization (EM) algorithm [136] embeds the impact of robot's uncertainty directly in a virtual map. A similar approximation can be done for particle-filter SLAM, which represents the belief over robot trajectories as a set of particles [2], [3], [37]. The integral in (10) will be now approximated by the weighted mean of all possible solutions (i.e., particles) [38].

The first term in (10) refers to the robot state entropy, which can be computed as a function of the posterior covariance log-determinant, assuming that it is an ℓ-dimensional Gaussian distribution with covariance $\boldsymbol{\Sigma}_r \in \mathbb{R}^{\ell \times \ell}$

$$\mathcal{H}\left[p(\boldsymbol{x}|\boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a})\right] = \frac{1}{2} \ln \left((2\pi e)^{\ell} \det\left(\boldsymbol{\Sigma}_r\right)\right). \quad (12)$$

On the other hand, the second term is the expected map's entropy, and its computation depends on the representation chosen. For instance, in landmark-based maps it can be computed in the same way as the robot's entropy, under the same assumption [137]. For discrete metric maps, and assuming cells independent from each other, it can be defined as [31]

$$\mathcal{H}\left[p(\boldsymbol{m}|\mathbf{x}, \boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a})\right] = -\sum_{c \in \boldsymbol{m}} \theta_c \log \theta_c \quad (13)$$

with $\theta_c = p(c)$ being the occupancy probability of cell c. This entropy measure has been used in both 2-D [28], [133] and 3-D OG maps [85], [88], [117]. More efficient approaches that only evaluate of cells in the robot's vicinity have been proposed in the context of particle-filter SLAM [42], [134], [138].

The most common metric to assess utility in active SLAM is not Shannon's entropy of the SLAM posterior, but its expected reduction. This utility function is known as MI [27], [33] and is defined as the difference between the entropy of the actual state and the expected entropy after executing an action, i.e., the information gain

$$\mathcal{I}(\boldsymbol{a}) \triangleq \underbrace{\mathcal{H}\left[p\left(\boldsymbol{x}, \boldsymbol{m}|\boldsymbol{h}\right)\right]}_{\text{current } \mathcal{H}} - \underbrace{\mathbb{E}\left[\mathcal{H}[p(\boldsymbol{x}, \boldsymbol{m}|\boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a})]\right]}_{\text{expected } \mathcal{H} \text{ for candidate } \boldsymbol{a}} \quad (14)$$

where expectation is taken w.r.t. $\hat{\boldsymbol{z}}$.

Kullback–Leibler divergence (KLD) or relative entropy [139] has also been used as utility function. KLD measures the change in the form of a pdf (as MI), but also how much its mean has translated [140]. It is defined as follows:

$$\mathcal{D}_{\text{KL}}\left(p_1|p_2\right) \triangleq \mathbb{E}\left[\log \frac{p_1(\boldsymbol{x})}{p_2(\boldsymbol{x})}\right] = \sum_{\boldsymbol{x}} p_1(\boldsymbol{x}) \log \frac{p_1(\boldsymbol{x})}{p_2(\boldsymbol{x})} \quad (15)$$

with $p_1(\boldsymbol{x})$ and $p_2(\boldsymbol{x})$ the prior and posterior distributions (as in MI) [141], or the estimated and true posteriors assuming the latter can be somehow approximated [42], [142], [143].

For OG maps, the three metrics above (entropy, MI, and KLD) ultimately rely on counting the number of cells in a map, being, thus, discrete and ill-suited for optimization techniques. To mitigate this issue, Deng et al. [87], [144] proposed a differentiable *cost-utility* function for both 2-D OG and voxel maps that can be used with continuous optimization methods (albeit the approach still assumes perfect robot localization).

In the context of information-theoretic planning, there exists a problem variation in which the uncertainty of only a subset of variables is reduced. The motivation comes from the fact that maximizing information of all variables does not always imply maximizing that of the subset of interest. This problem variation has been referred to as focused active inference [145]. In general, focused active inference is more computationally intensive than the standard case, since it requires marginalization of the (posterior) Fisher information matrix (FIM) via, e.g., Schur complement. Kopitkov and Indelman [146], [147] presented a method based on the matrix determinant lemma that does not require the posterior covariance to calculate entropy considering both the unfocused (entropy over all variables) and focused (entropy over a subset of variables) cases.

C. Theory of Optimal Experimental Design (TOED)

There exists a second group of utility functions built upon optimal design theory (TOED) that tries to quantify uncertainty directly in the task space (i.e., from the variance of the variables of interest). Unlike information-theoretic metrics that target binary probabilities in the grid map, task-driven metrics apply to Gaussian variables. Following TOED, a set of actions to execute in active SLAM will be preferred over another if the covariance of the joint posterior is smaller, i.e., the posterior covariance matrix, $\boldsymbol{\Sigma}$, has to be minimized. In order to compare matrices associated to different candidates, several functions—known as *optimality criteria*—have been proposed, such as the trace (originally known as A-optimality) [148], its maximum/minimum eigenvalue (E-optimality) [149], or its determinant (D-optimality) [150]. The latter was often disregarded in active SLAM because its traditional formulation did not allow for checking task completion and generated precision errors $(\det(\boldsymbol{\Sigma}) \to 0$ rapidly when there are low-variance terms) [34], [140]. However, Carrillo et al. [24] showed these problems can be solved using Kiefer's formulation of D-optimality [151], thus re-establishing the latter as an effective measure of uncertainty for active SLAM.

On the basis of TOED, Kiefer [151] proposed a family of mappings $\|\boldsymbol{\Sigma}\|_p : \mathbb{R}^{n \times n} \to \mathbb{R}$, parametrized by a scalar p

$$\|\boldsymbol{\Sigma}\|_p \triangleq \left(\frac{1}{n} \text{trace}(\boldsymbol{\Sigma}^p)\right)^{\frac{1}{p}} \quad (16)$$

which can be particularized for the different values of p and expressed in terms of the eigenvalues of $\boldsymbol{\Sigma}$, $(\lambda_1, \ldots, \lambda_\ell)$, by leveraging the properties of the matrix power

$$\|\boldsymbol{\Sigma}\|_p = \begin{cases} \left(\frac{1}{n} \sum_{k=1}^{n} \lambda_k^p\right)^{\frac{1}{p}}, & \text{if } 0 < |p| < \infty \\ \exp\left(\frac{1}{n} \sum_{k=1}^{n} \log(\lambda_k)\right), & \text{if } p = 0 \end{cases}. \quad (17)$$

In essence, utility functions are functionals of the eigenvalues of $\boldsymbol{\Sigma}$. The boundary cases $p = \{0, \pm\infty\}$ and $p = \pm 1$ result in the following four modern optimality criteria.

1) *T-optimality criterion* ($p = 1$) captures the average variance

$$T\text{-opt} \triangleq \frac{1}{n} \sum_{k=1}^{n} \lambda_k. \quad (18)$$

2) *D*-optimality criterion ($p = 0$) captures the volume of the covariance (hyper) ellipsoid

$$D\text{-opt} \triangleq \exp\left(\frac{1}{n}\sum_{k=1}^{n}\log(\lambda_k)\right). \quad (19)$$

3) *A*-optimality criterion ($p = -1$) captures the harmonic mean variance, sensitive to a lower-than average value

$$A\text{-opt} \triangleq \left(\frac{1}{n}\sum_{k=1}^{n}\lambda_k^{-1}\right)^{-1}. \quad (20)$$

4) *E*-optimality criterion ($p \to \pm\infty$) captures the radii of the covariance (hyper) ellipsoid

$$E\text{-opt} \triangleq \min(\lambda_k : k = 1,\ldots,n) \quad (21)$$

$$\tilde{E}\text{-opt} \triangleq \max(\lambda_k : k = 1,\ldots,n). \quad (22)$$

Optimality criteria were first used in active SLAM by Feder et al. [26], where utility was computed as the area of the covariance ellipses describing the uncertainty in the joint posterior. Since then, many active SLAM methods based on TOED have been proposed, mostly based on T-opt [34], [39] and, recently, D-opt [61], [123]. Even so, IT-based methods remain the most popular. Note that both the map and robot uncertainties must be described by a covariance matrix $\mathbf{\Sigma} \in \mathbb{R}^{n \times n}$, either by using a full covariance matrix in landmark-based representations (i.e., $n \gg \ell$) or by including the effect of the map's uncertainty in $\mathbf{\Sigma}_r$ (and thus $n = \ell$) [152]. *Monotonicity*. One of the most important assumptions in active SLAM is that uncertainty increases as exploration takes place. However, the seminal work in [153] notes how monotonicity is lost for some utility functions under certain conditions, concluding that only D-opt guarantees this property and is thereby the only appropriate utility function for this task. Kim and Kim [154] and Rodríguez-Arévalo et al. [155] demonstrate, however that rather than on the utility function chosen, monotonicity depends on how the error and uncertainty are represented. In [155], the authors prove that only differential representations guarantee monotonicity for all utility functions. In summary, representation of uncertainty is a key issue in active SLAM, since certain representations do not guarantee its monotonicity property during exploration, and thus, may lead to incorrect decisions.

D. Graphical Structure of the Problem

Quantification of uncertainty via scalar mappings of the covariance matrix may be a computationally intensive task, mostly due to the fact that the covariance is a large and dense matrix. Therefore, most works resort to reasoning over the FIM, i.e., the inverse of the covariance, which is generally sparser. Still, their evaluation is expensive, especially for large state spaces. To circumvent this issue, some works have proved that analyzing the connectivity (i.e., Laplacian) of the underlying pose-graph in active graph-SLAM is equivalent to computing optimality criteria. The link between graph and optimum design theories can be traced back to Cheng [156], who related the number of spanning trees of concurrence graphs with D-optimal incomplete block designs. Khosoussi et al. [157] showed that classical D- and E-opt are related to the number of spanning trees of the SLAM pose-graph and its algebraic connectivity, respectively, for the case of 2-D graph-SLAM with constant uncertainty along the trajectory. In [158] and [159], these results

are extended to the $\mathbb{R}^n \times \mathrm{SO}(n)$ synchronization problem, and also relate T-opt to the average node degree of the graph. Placed and Castellanos [81], [160] studied the general active graph-SLAM problem formulated over the Lie group $\mathrm{SE}(n)$; showing the existing relationships between modern optimality criteria of the FIM and connectivity indices when the edges of the pose-graph are weighted appropriately, and reporting substantial reductions in computation time. These results have been used in coverage problems [62], multirobot exploration [128], active visual SLAM [66], or to develop a *stopping criterion* [161].

The graph structure of the problem has also been recently exploited in conjunction with IT utility functions. Kitanov and Indelman [162] relate the number of spanning trees of the graph to entropy (which ultimately depends on the covariance determinant) and its node degree to Von Neumann entropy. The latter has been also applied to the focused case, thus relating the graph topology to the marginalized FIM [163].

V. STAGE 3: ACTION SELECTION AND EXECUTION

Once every possible destination has an associated utility value, the last stage of active SLAM involves the selection of the optimal destination. This can be formulated as an optimization problem w.r.t. the set of actions to reach every possible goal location, cf. (9). When the set of candidate destinations is discrete (and typically consists in a handful of options), the solution of the optimization can be obtained via enumeration [49], [74], [114]. For the case of TOED-based utility functions, it will be a minimization or maximization problem depending on whether the covariance ($\mathbf{\Sigma}$) or the FIM ($\mathbf{\Phi}$) is analyzed. Since $\mathbf{\Sigma} = \mathbf{\Phi}^{-1}$ and $\|\mathbf{\Sigma}\|_p = (\|\mathbf{\Phi}\|_q)^{-1} \forall p$ with $q = -p$, the optimization problem is

$$\boldsymbol{a}^\star = \arg\min_{\boldsymbol{a} \in \mathcal{A}} \|\mathbf{\Sigma}\|_p = \arg\max_{\boldsymbol{a} \in \mathcal{A}} \|\mathbf{\Phi}\|_q \quad (23)$$

where $\|\cdot\|_p$ refers to Kiefer's optimality criteria, see (16).

Information-based utility functions will seek to minimize entropy (or, equivalently, to maximize MI). Following [40], the optimal set of discrete actions can be found as

$$\boldsymbol{a}^\star = \arg\max_{\boldsymbol{a} \in \mathcal{A}} \mathcal{I}_G = \arg\min_{\boldsymbol{a} \in \mathcal{A}} \mathcal{H}\left[p(\boldsymbol{x}, \boldsymbol{m}|\boldsymbol{h}, \hat{\boldsymbol{z}}, \boldsymbol{a})\right]. \quad (24)$$

In any case, after selecting the most informative destination, it all comes down to navigating to it using, e.g., sampling-based planning methods as RRT [164], probabilistic road maps (PRM) [165], or their asymptotically optimal variants [166]. Note that despite selecting the optimal destination among a discrete set of candidates, the executed path to reach it rarely represents an optimal solution for the original problem (9); this suboptimality is caused by decoupling the problem into first computing and evaluating a set of goal locations, and then computing a path to one of these goals.

VI. BSP AND CONTINUOUS-SPACE OPTIMIZATION

As a potential solution to the suboptimality induced by classical decoupled approaches, there exists a second family of methods in which the future trajectory of the robot is directly optimized. These methods represent an alternative solution to the modular scheme and may be divided into two categories, depending on whether they discretize the action space or not. The first category relies on path planning algorithms to generate a discrete set of candidate paths toward the unknown space, in

order to later evaluate their utility. Works from Oriolo et al. [167] and Freda et al. [168] are among the first to apply these algorithms for exploration, evaluating robot configurations inside the not-previously-sensed free space. In contrast to discrete frontier optimization that compares utility only at candidate locations, these methods evaluate it over the paths to reach them, guaranteeing that the path to execute is optimal among the considered subset. Bonetto et al. [68], [169] go one step further and optimize exploration in all three steps of modular approaches, considering not only the destination and the path to reach it, but also its execution.

On the other hand, globally optimal solutions have been considered under the umbrella of continuous-state POMDPs. Despite their resolution would ideally require to compute a policy over the infinite-dimensional space of posteriors of the joint state space [170] and computing an exact solution is known to be intractable in general [171], active SLAM as a continuous-state POMDP can be approximately solved under the frameworks of BSP or optimal control. Such optimization techniques require a continuous utility function, which can be obtained directly from complex continuous representations of the environment [53] or inferred from discretized representations. For example, Vallvé and Andrade-Cetto [133] compute a dense entropy field from the posteriors' evaluation over the discretized configuration space.

A. Belief-Space Planning

Continuous-domain BSP optimizes the future trajectory of the robot without discretizing the action space, but rather performing a continuous optimization. Bai et al. [172] and Kontitsis et al. [143] used sampling-based methods to maximize an objective function that rewards uncertainty reduction and goal achievement. Platt et al. [32] applied linear quadratic regulation to compute locally optimal policies. Van Den Berg et al. [120] relax the assumption that future observations are consistent with the current robot pose belief (ML observations). Indelman et al. [43] extend [120] to the case where the belief describes both robot poses and unknown landmarks in the environment, while also modeling missed observations. Porta et al. [173] generalized value iteration to continuous-state POMDPs while assuming state-dependent reward functions. Van den Berg et al. [170] present a highly efficient method for solving continuous POMDPs in which beliefs can be modeled using Gaussian distributions over $\mathcal{S}$. Prentice and Roy [174] developed a belief-space variant of the PRM algorithm called the belief road map (BRM), incorporating predicted uncertainty of future position estimates into the planning process. Valencia et al. [175] contributed a pose-SLAM path-planning approach that leverages the BRM to find a path to the goal with the lowest accumulated pose uncertainty.

B. Active SLAM as Optimal Control

Converting a POMDP formulation of active SLAM into an equivalent continuous-space MDP, as discussed in Section II, leads to a stochastic optimal control problem in general. Depending on the transition and observation models, noise distribution, and the reward function, the problem may be simplified further. Le Ny et al. [176] and Atanasov et al. [9] showed that when the transition and observation models are linear in the state s and the noise is Gaussian, then the time evolution of the belief state b_t may be obtained by the Kalman filter and the covariance is *independent* of the measurement realizations. If the reward function ρ depends only on the covariance, as for the MI, active SLAM reduces from a stochastic to a deterministic optimal control problem. Deterministic optimal control problems are easier to solve, and techniques, such as linear–quadratic–Gaussian (LQG) regulation [177] or search-based [9], [178], and sampling-based [179], [180], [181] motion planning are applicable. If the assumptions necessary for the deterministic reduction cannot be satisfied, the stochastic active SLAM problem may be solved by obtaining an open-loop control sequence under deterministic dynamics first, followed by a closed-loop feedback policy, under stochastic dynamics linearized around open-loop trajectory [182].

In the presence of state or action constraints, the optimal control formulation of active SLAM can be approached using differential dynamic programming (DDP) or model predictive control (MPC). Rahman and Waslander [183] introduced an augmented Lagrangian formulation of iterative LQG, which captures belief-state constraints via a penalty function. The approach iterates between iLQG trajectory optimization in an unconstrained stochastic optimal control problem and Lagrange multiplier updates for the penalty function. This and several other works [177], [183], [184] develop differentiable formulations of sensor field-of-view constraints amenable to gradient-based optimization. Carlone and Lyons [185] split the environment into convex regions and formulate the problem using mixed-integer programming. Chen et al. [62] employed a spectrahedral description of the convex hull of the space of orientations and relax nonconvex obstacle constraints using a convex half-space representation.

Striking a suitable balance between exploration and exploitation in active SLAM is challenging because the effects of potential future loop closures are not easy to capture in the predicted evolution of the belief $b_t(s_t)$. Leung et al. [39], [186] introduced attractor states to guide the robot based on three modes (explore, improve map, and improve localization), determined using uncertainty thresholds. Attractor states were combined with a right-invariant extended Kalman filter in [187] to achieve active range-bearing landmark-based SLAM.

VII. DL-BASED APPROACHES

Advances in DL have created new opportunities in using neural networks to solve active SLAM; these techniques follow a completely different scheme, circumventing the split into three stages that characterizes modular approaches. Usually, goal identification is not required due to the chosen action set, and utility computation and selection of the best action are both embedded in the network. In this section, we particularly focus on DRL methods for autonomous robotic exploration and discuss the design of the state, action, and reward spaces, as well as the problems of partial observability, generalization, and the necessity for training environments.

A. Deep Reinforcement Learning

A question that arose in the early work on learning-based active SLAM was which type of learning was suitable for this decision-making problem, in which
1) agents must directly learn from interaction with the environment;
2) states may not be fully observable;

3) policies have to generalize to other scenarios in which a priori knowledge is nonexistent.

This premise soon led the community to explore DRL, building on existing methods that approached active SLAM with RL [188] and using neural networks to represent the policies or value functions. Within DRL, model-free techniques have been the center of attention, although isolated approaches that combine them with model-based learning do exist [189]. Methods based on supervised learning can also be found in the literature [190], [191], although they are a minority. Contrary to model-based active SLAM, the computational effort in DRL approaches is mostly confined to the training phase, while the testing phase reduces to a forward pass on the network. However, the behavior depends entirely on the model learned from training data, thus limiting its generalization to novel operational conditions.

The great success of the work from Mnih et al. [192] boosted the research in model-free DRL and several value- and policy-based methods emerged shortly after. The behavior of deep Q-networks [192] improves using the double [193] and double-dueling [194] architectures. Actor–critic techniques combine both value-iteration and policy gradient methods, e.g., deep deterministic policy gradient [195], asynchronous advantage actor–critic [196]. See [197] and [198] for a survey on the methods. Although these strategies were initially proposed for different decision-making problems (e.g., video-games), they have been applied to robotic exploration.

B. On the Reward Function Design and the Action Set

Tai and Liu [199] are among the first to employ DRL for robotic exploration in simulation environments, extracting the next best actions to execute from raw observations using a two-layer Q-network. Convergence to policies valid in more complex and previously unseen scenarios is achieved in [121], [200] with parallel architectures. In any case, the abovementioned works use purely extrinsic reward functions (i.e., by instrumenting the environment), which ultimately addresses the obstacle avoidance problem rather than active SLAM [123]. As a response, the notions of motivation and curiosity [201] were exploited to design intrinsic rewards, giving origin to *curiosity-driven* methods that motivate agents to visit unknown configurations [202]. Chen et al. [203] and Chaplot et al. [56] proposed holistic, open-source approaches that employ a coverage reward to explore complex 3-D simulation environments. The detailed study in [203] shows the benefits of pretraining and combining inputs from different sources. Similarly, the idea of uncertainty minimization led to *uncertainty-aware* approaches. This is the case of [204] that encourage the visit of high-covariance states, and [20], [122] where the reward encodes the belief accuracy. All these methods are publicly available except active target localization. Many of the DRL-based methods, including all of the above, aim to directly generate optimal control commands, either discrete [203] or not [205]. They represent end-to-end solutions in which the safe navigation task is embedded into the network and, therefore, do not require planning and the SLAM estimates.

True uncertainty metrics inherited from classic theories have also been introduced in the reward function design, seeking more robust foundations. The robot's D-opt is incorporated in [123] and T-opt of virtual landmarks in [58], whereas the map's MI is used in [57] and [59]. Agents trained under this new perspective

perform active SLAM in complex scenes, albeit only targeting location or mapping uncertainties. Designing effective reward functions that account for both is still an open problem. In addition, this new family of methods has promoted the use of learning as a part of the solution rather than a replacement to well-established planning algorithms. Utilizing planning and learning together, may make policies easier to learn, generalize better and transfer across platforms. In this vein, Niroui et al. [57] and Chen et al. [58] employed DRL to extract the best candidate among previously-detected frontiers, thereby creating a link with modular approaches. Li et al. [59] and Lodel et al. [206] used nearby sampled locations instead, but they also leave the motion planning task out of the scope of learning. Chaplot et al. [56] used different policies to infer long-term (i.e., frontiers) and short-term (i.e., control commands) goals, linked through a model-based trajectory planner.

C. Partial Observability and Generalization

Partial observability and generalization are two inherent and often-forgotten concepts in active SLAM. First of all, the uncertainty about the observations and actions taken, and the limited observations make the problem not fully observable. Consequently, agents are unable to distinguish their own true state based on single observations, and learned policies are bound to be suboptimal [207]. Mirowski et al. [200] alleviated this by expanding the network inputs with previous observations and rewards. Hausknecht and Stone [208] demonstrated that recurrent architectures can also handle partial observability, teaching agents to learn about previous data on their own. Long short-term memory units are used for robotic exploration in [57], [200], [203], and Karkus et al. [189] embed the computation structure of the belief (and thus, the history) in a recurrent neural network.

The second element intrinsic to active SLAM is the lack of prior knowledge of the environment. Learning policies that generalize to unseen scenarios is, therefore, crucial, and currently represents a key limiting factor for learning-based methods. Overfitting can be mitigated by expanding the sample space (e.g., using random starting locations [57], [204], considering noise in the observations [209]) or by using sparser network inputs [207]. For example, agents trained in [121] and [210] learn policies generalizable to real environments after reducing sensory data to a sparse range input. Similarly, Shi et al. [211] specifically use sparse range measurements to reduce the simulation-to-reality (*sim-2-real*) gap. Lodel et al. [206] improved generalization by feeding the network with egocentric limited observations, following [203]. Chen et al. [58] leverage graph neural networks, in which the inputs are already compressed representations. The task of transferring trained agents to real scenarios is still an open research problem, and few efforts have been made in this direction [59], [121], [211].

D. Training Environments

The use of DRL introduced a major challenge during training: the need of a simulation environment to acquire data online. Unlike supervised methods, training with offline data is not possible and real-world training seems infeasible. To overcome this problem, some works use their own simplified simulation scenarios, thus limiting the network inputs to ground-truth data or range perfect observations. To use more realistic data that

bridge the gap from simulation to physical robots, more complex simulators need to be used in training.

Stage [212] is one of the simplest engines used in the literature [57], although it restricts perception to two-dimensional bitmapped environments. *Gazebo* [213] is a much more complete simulator, which allows for 3-D simulations, realistic rendering, visual sensors, etc. In addition, it is tightly integrated into the widespread Robotic Operating System (ROS), which makes its use commonplace [121], [123], [199]. *CoppeliaSim/V-REP* [214] also allows for online mesh manipulation, but it is not an open-source solution and is less integrated into ROS, limiting its adoption. Combination of a physics engine (i.e., robot motion and sensor models) with a DRL framework is not always straightforward. Zamora et al. [215] present a powerful framework by integrating the RL toolkit OpenAI Gym [216] with ROS and Gazebo.

In contrast to the abovementioned platforms, initially designed for robotics and later adapted to DRL, there is a second family of simulators born in the age of AI. They tend to prioritize training speed over the breadth of simulation capabilities. *DeepMind Lab* [217] allows agents to move discretely in low-textured, game-like scenarios, and provides access to a visual sensor and velocity. *Habitat-Sim* [218] takes a leap forward by supporting physics simulation and different robot and visual sensor models. More interestingly, it has the powerful capability of rendering simulation environments from image datasets, e.g., Replica [219]. *iGibson* [220] also provides fast visual rendering and physics simulation, and includes simulation of lidar and optical flow sensors. The ROS ecosystem is already integrated in [220], whereas Savva et al. [218] required the use of external libraries. Despite their potential, none of these platforms has yet been used for DRL in the context of active SLAM.

VIII. MULTIROBOT ACTIVE SLAM

The active SLAM problem can be extended to a multiagent setting, where n robots optimize their sensing trajectories collaboratively to estimate a common map $m \in \mathcal{M}$ of the environment. Each robot has its own state space $\mathcal{X}_i$ and action space $\mathcal{A}_i$. Applying an active SLAM algorithm to the joint state space $\mathcal{S} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_n \times \mathcal{M}$ and joint action space $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_n$ can generate desirable behavior but becomes computationally infeasible as the number of robots increases because the complexity of centralized algorithms scales exponentially with n [78]. Such algorithms also require collecting all robot measurements and performing joint optimization at a centralized server before communicating the planned actions back to the individual robots. If the robot team is small and connectivity is maintained at all times, centralized algorithms can be used to plan all robot trajectories simultaneously. For example, Charrow et al. [221] achieve multirobot target tracking by maximizing the MI between the target location and range-only observations over a set of motion primitives. However, larger teams with intermittent communication and limited onboard computation require decentralized algorithms, where individual robots solve smaller instances of the active SLAM problem, or fully distributed algorithms, where the robots exchange information only with their neighbors. Kantaros et al. [181], [222] proposed an informative planning[2] technique, which constructs random trees of control sequences and is particularly simple to distribute. The algorithm scales to very large numbers of sensors and targets and is probabilistically complete and asymptotically optimal.

A particularly important instance of the problem is collaborative multirobot exploration, where the robots aim to coordinate how to efficiently explore different regions of the environment. Early works such as [105], [223] present an approach for choosing appropriate frontiers, while simultaneously taking into account their utility and the cost of reaching them. Each time a target point is assigned to some specific robot, the utility of the unexplored area visible from that frontier is reduced. This mechanism is used to assign different frontiers to different robots. Colares and Chaimowicz [132] developed a decentralized multirobot formulation of the classical frontier-based exploration method. The authors use an objective function that captures the frontier entropy and distance, and a robot coordination factor that penalizes regions that other robots are already exploring. Atanasov et al. [78] considered a multiagent active information acquisition problem, in which an information measure is maximized over a discrete space of agent trajectories, and proposed a decentralized planning scheme using coordinate descent in the space of agent trajectories. Schlotfeldt et al. [224] introduced an anytime search-based planning formulation that progressively reduces the suboptimality of the multiagent plans while respecting real-time constraints. Instead of using search-based planning, Ossenkopf et al. [225] generated candidate robot actions using RRT*. The sampling is biased to prioritize exploration, map improvement, or localization improvement. The map and robot state entropy is evaluated along the planned trajectories in two stages: short-horizon exact computation using filter updates, and long-horizon approximation using predicted loop closures. Lauri et al. [226] introduced a decentralized ρ-POMDP, allowing the specification of an information-theoretic objective. The authors show that a multiagent A* algorithm that searches the joint policy space can be applied to belief-dependent rewards to achieve cooperative target tracking with periodic communication. Hu et al. [205] designed a hierarchical control approach for cooperative exploration, combining a high-level region-assignment layer and a low-level safe-navigation layer. The former uses dynamic Voronoi partitions to assign different regions to individual robots; the latter achieves collision-free navigation to successive frontier points using DRL.

Another important instance is collaborative multirobot active estimation, where the goal is to seek actions that actively reduce the uncertainty over relevant random variables. For instance, Kontitsis et al. [143] developed a multirobot active SLAM method that uses a relative entropy optimization technique [227] to select trajectories which minimize both localization and map uncertainties. Indelman [228] develops a collaborative multirobot BSP framework, which incorporates reasoning about future observations of environments that are unknown at planning time. That approach has been extended in [229] to a decentralized setting. Best et al. [230] proposed the *self-organizing map* algorithm, considering the problem of multirobot path planning for active perception and data collection tasks. Chen et al. [128] leverage graph connectivity indices and their relationship to optimality criteria to achieve multirobot active graph-SLAM. Each robot aims to improve the pose graphs of the other agents by sharing its observations when it moves near areas where they have low connectivity.

---

[2]Informative path planning can be considered a generalization of active SLAM to include objectives beyond the quality of localization and mapping, e.g., for target tracking or environmental monitoring.

IX. Open Research Questions

Active SLAM still requires much research in order to support the deployment of fully autonomous robots in complex environments. Many are the challenges and research fields involved, so cooperation between them is crucial to achieve real-world impact. In this section, we present some of what we consider the most important research questions. Although some of them are long-known challenges and are already under intense investigation, others have not received such attention.

A. Prediction Beyond Line-of-Sight

Resolution of active SLAM relies on fast and precise predictions of future states for the variables of interest. The accurate prediction of the scene and robot pose after executing a set of candidate actions can be the difference between making the right decision or not. The expected sensed space and the resulting map representation have traditionally been predicted using a sensor model together with ray-casting techniques [51], [105]. Recent related work, however, addresses the problem of scene completion and occupancy anticipation from a DL perspective. Fehr et al. [231] used a neural network to augment the measurements of a depth sensor and Ramakrishnan et al. [232] directly predict augmented OG maps beyond the sensor's field-of-view using autoencoders (AE). Rather than using raw sensor measurements, Katyal et al. [233] and Hayoun et al. [234] extend an input OG map beyond the line-of-sight also using AE. Shrestha et al. [235] predict maps of occupancy probabilities instead with variational AE. Dai et al. [236] perform scene prediction over 3-D SDF-based maps. All these methods seem promising for fast and precise online map prediction beyond line-of-sight, although their integration into active SLAM is yet to be done and brings with it numerous challenges. How does scene prediction behave in unstructured environments? How to account for uncertainty? Is measurement prediction more reliable and informative than map prediction? How to predict the effect of only a certain set of nonmyopic actions in the map rather than augmenting the whole scene? Regarding the latter, [237], [238], and [239] subordinate predictions to candidate actions.

On the other hand, the robot state is straightforwardly estimated using motion models or path planners. However, the prediction of its associated uncertainty is not trivial and requires more attention. Work from Asraf and Indelman [239] is among the very few efforts made to combine data-driven scene prediction with BSP. In addition, they use the predicted observations to forecast the posterior uncertainty over the robot trajectory. Besides the robot's movement, it is the appearance of loop closures (exploitation) that significantly affects the new states' uncertainty, thus making its forecast critical. Despite some isolated works have partially studied this problem [29], [240], it still remains as an open challenge.

B. From Active SLAM to Active Spatial Perception

Most active SLAM approaches reason over geometric representations of the environment (e.g., OG maps). However, when we explore new environments as humans, we are mostly interested in semantic elements of the environment (e.g., presence of objects, rooms) rather than the shape of the environment *per se*. Modern SLAM systems are now able to build 3-D metric-semantic maps in real-time from semantically labeled images, see [96] and the references therein. These maps include both occupancy information and semantic labels of entities (e.g., chairs, tables, humans, etc.) in the environment. Very recent work goes even further and develops spatial perception systems that infer hierarchical map representations, in the form of 3-D scene graphs [103], [241], [242]. They symbolize high-level representations of an environment that capture from low-level geometry (e.g., a 3-D mesh of the environment) to high-level semantics (e.g., objects, people, rooms, buildings, etc.). While there is a growing amount of work in estimating these high-level representations from sensor data, their use in active SLAM is still uncharted territory. Very early effort in this direction includes the work from Ravichandran et al. [243], which focuses on object search using 3-D scene graphs.

Active metric-semantic information acquisition, or active spatial perception, has the potential to impact many aspects of robot autonomy:
1) by leveraging semantic knowledge, a robot can more effectively predict unseen space (e.g., predict the presence of rooms or objects in each room);
2) the use of semantics can further enhance the SLAM performance (e.g., allowing for novel loop closure detection methods [242]);
3) hierarchical representations may enable novel and more computationally efficient planning methods.

However, each opportunity comes with many open research questions, for instance: How to quantify uncertainty over metric-semantic or even hierarchical scene representations? How to leverage hierarchical structures to improve computation? How to perform spatial prediction in hierarchical representations?

C. Robust Online BSP and Active SLAM

Another key aspect is data association, i.e., association between measurements and the corresponding landmarks (or other entities in the map representation). In perceptually aliased and ambiguous environments, determining the correct data association is challenging, and incorrect associations may lead to catastrophic failures. The research community has been investigating approaches for robust perception to allow reliable and efficient operation in ambiguous environments (see, e.g., [244], [245], [246], [247], [248], [249]). Yet, these approaches focus on inference (rather than planning), i.e., actions are given. Only recently, ambiguous data association was considered also within BSP and, in particular, active SLAM. Pathak et al. [250] incorporate, for the first time, reasoning about future data association hypotheses within a BSP framework, enabling autonomous hypotheses disambiguation. Another related work in this context is [251] that also reasons about ambiguous data association in future beliefs while utilizing the graphical model presented in [248]. A first-moment approximation to Bayesian inference with random sets of targets, known as the probability hypothesis density filter, has been successfully applied to active target tracking problems [252]. However, explicitly considering all possible data associations leads to an exponential growth of the number of hypotheses, and determining the optimal action sequence quickly becomes intractable. Shienman and Indelman [253] recently presented an approach that utilizes only a distilled subset of hypotheses to solve BSP problems while reasoning about data association and providing performance guarantees considering a myopic setting. Nevertheless, BSP and active SLAM in these challenging settings remain open problems. More generally, finding an appropriate simplification of the

original BSP or active SLAM problem, which is easier to solve, with no, or bounded, loss in the performance, is an exciting and novel direction [253], [254], [255], [256].

D. Reasoning in Dynamic and Deformable Scenes

One of the most common assumptions in active SLAM is to consider the environment static—or slightly dynamic, at best. Real scenes, however, contain moving agents most of the times, and even deformable elements (e.g., clothes, water). Handling these elements would greatly impact the robot's autonomy, its reasoning ability and awareness, and would facilitate its deployment in real-world scenarios.

The study of dynamic environments has long been a topic of interest for the path planning [257] and the SLAM [258] communities; but its investigation in the context of active SLAM has been typically restricted to the action execution step (i.e., replanning) [259], [260]. However, many other aspects emerge when reasoning with dynamic elements: How to foresee their effects in planning? How to integrate them in the utility function? How to maintain a lightweight representation?

Nonrigid environments present an even greater challenge. Planning for mobile robots in deformable environments started receiving some attention a couple of decades ago [261], [262]. Medical applications have also stimulated progress on SLAM in deformable environments [263], [264]. However, to date, no efforts have been made toward developing a deformable active SLAM framework. We believe this is partly due to the unavailability and complexity of simulators for mobile robots in deformable environments, and partly due to the difficulty in extending the current map representations to deformable scenes. Given the importance of obtaining accurate robot trajectory estimates toward mapping deformable environments, active SLAM can play a major role in this area.

E. Toward Meaningful and Autonomous Stopping Criteria

Unlike with coverage and exploration in known environments, determining the moment in which the task of active SLAM is complete is nontrivial. Performing active SLAM is known to be a computationally expensive process: a vast amount of resources is required to estimate and optimize utility online, thereby conditioning the execution of other tasks; therefore, it is crucial to understand when such process can be considered complete and other tasks can be prioritized. Cadena et al. [5] already identified this issue as an open research question, but little research has been done on the topic. Even recent active SLAM works still rely on traditional temporal [51], [81] or spatial [62], [265] constraints to decide when exploration has terminated. These metrics, however, cannot be used in truly unknown environments nor do they assert task completion (see [161]). The use of TOED-based metrics has been identified as a promising tool [5], [36], [161] to determine when a given exploration strategy is no longer adding information. Nevertheless, many questions remain to be answered: How to guarantee task completion? How to transition between exploration strategies? Also, the advent of DRL approaches raises a new question: when to stop training?

F. Reproducible Research in Active SLAM

The increasing attention toward active SLAM creates the need for new benchmarks to objectively evaluate new findings against existing research. This has long been a challenge in the robotics community [266], where real-life robotics experiments are often difficult to replicate across research groups. In related problems, such as SLAM, static datasets are commonly used for benchmarking. However, in active SLAM, the agent must interact with the environment, making the use of datasets impractical. In recent years, a significant effort has been made in robotics to address challenges in benchmarking [267] and reproducibility [268]. Despite these efforts, such benchmarks are still lacking in active perception.

Typically, in active SLAM, researchers select a set of scenarios (e.g., platform, task, and environment) representative of the desired application, and experiments are conducted in simulation via customized simulators or in the real world via specialized hardware. While such an evaluation is adequate for validation, the specified scenario may not be general enough or sufficiently specified to be reproduced. Consequently, one-to-one comparisons are rarely made between approaches. While targeting more general embodied agents, several open-source datasets [269] and simulators [218], [265], [270] show promise for active SLAM research. Also, open-source frameworks (see Table I) make the comparison and testing of new algorithms straightforward, only by modifying the decision-making portion. While some works take advantage of these simulators and datasets [56], establishing a proper methodology for evaluating active SLAM when it comes to generalization from simulation to the real world remains an open question. Besides, there is a dire need to establish adequate performance metrics for active SLAM that go beyond commonly-used exploration time and coverage. Improving the quality of estimates is the main objective of active SLAM, and should, therefore, be measured.

G. Practical Applications

Although active SLAM methods have practical relevance in many real-world problems, such as search and rescue, where constructing a sound representation of the environment is time critical, very few practical implementations and deployments of active SLAM have been described in the literature. Walter et al. [271] proposed a partially autonomous system for underwater ship hull inspection. Kim and Eustice [119] deployed a complete active SLAM system. Palomeras et al. [54], [85] reported the autonomous exploration of complex underwater environments for environmental preservation purposes. Fairfield and Wettergreen [240] investigated terrestrial applications and autonomous mapping of abandoned underground mines. A roughly similar application but in the archaeological context of catacomb exploration is presented in [272]. Strader et al. [22] reported experiments of active perception in a Mars-analogue environment. Finally, assistive mapping examples for office-like environments can be found in [30], [273], and [251]. Aerial applications of active SLAM are significantly less common. Chen et al. [62] proposed an MPC framework to address coverage tasks while maintaining low uncertainty estimates.

Overall, there are very few reports of field experiments involving active SLAM systems. Besides, by 2022, there is a large imbalance between the patents using the terms SLAM and active SLAM,[3] about 39 000 for the former and 31 for the latter. This indicates that the technology readiness level of active

---

[3]We used "simultaneous localization and mapping" after:priority:19920101, and "active slam" OR "active simultaneous localization and mapping" after:priority:19920101 as queries search in the Google patents search platform.

SLAM is not in a deployment phase but in early development. Furthermore, it raises the question of whether active SLAM is important for all applications or whether human supervision is still preferred. Among the roadblocks preventing the transition from theory to applications (including the challenges mentioned in the previous sections), we also remark that the high computational complexity of active SLAM often clashes with application constraints, e.g., the low computational budget available on aerial robots.

X. Conclusion

The active SLAM problem, which consists in actively controlling a robot such that it can estimate the most accurate and complete model of the environment, has been a topic of interest in the robotics community for more than three decades, and is now receiving renewed attention—also thanks to the novel opportunities offered by learning-based methods. Despite the role of active SLAM in many applications, the disparity and lack of unification in the literature has prevented the research community from providing a cohesive framework, bringing algorithms to maturity, and transitioning them to real applications. In this article, we take a step toward this goal by taking a fresh look at the problem and creating a complete survey to serve as a guide for researchers and practitioners.

In particular, we present a unified active SLAM formulation under the umbrella of POMDPs, highlighting the most common assumptions in the literature. Then, we discuss the modular resolution scheme, which decouples the problem into goal identification, utility computation, and action selection. We delve into each stage, reviewing the most important theories and presenting state-of-the-art techniques. We then review alternative approaches that have drawn great interest and have undergone major advances in recent years, including (continuous) BSP and learning-based approaches. Finally, we discuss relevant work in multirobot active SLAM. Besides discussing the historical evolution and current trends in active SLAM, we also identify the most relevant open challenges in this field. These include prediction beyond line-of-sight and active spatial perception, among others. We also emphasize the need for a unified formulation and evaluation metrics that allow for direct comparison between works. Reproducibility and benchmarking need to be addressed for this field to mature and achieve real-world impact.

References

[1] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*. Cambridge, MA, USA: MIT Press, 2011.

[2] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics* (Intelligent Robotics and Autonomous Agents). Cambridge, MA, USA: MIT Press, 2005.

[3] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robot. Autom. Mag.*, vol. 13, no. 2, pp. 99–110, Jun. 2006.

[4] G. Grisetti, R. Kummerle, C. Stachniss, and W. Burgard, "A tutorial on graph-based SLAM," *IEEE Intell. Transp. Syst. Mag.*, vol. 2, no. 4, pp. 31–43, Winter 2010.

[5] C. Cadena et al., "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, Dec. 2016.

[6] R. Bajcsy, "Active perception vs. passive perception," in *Proc. IEEE Workshop Comput. Vis.*, 1985, pp. 55–62.

[7] C. K. Cowan and P. D. Kovesi, "Automatic sensor placement from vision task requirements," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 3, pp. 407–416, May 1988.

[8] J. Aloimonos, I. Weiss, and A. Bandyopadhyay, "Active vision," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 333–356, 1988.

[9] N. Atanasov, J. Le Ny, K. Daniilidis, and G. Pappas, "Information acquisition with sensing robots: Algorithms and error bounds," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 6447–6454.

[10] R. Bajcsy, "Active perception," *Proc. IEEE*, vol. 76, no. 8, pp. 966–1005, Aug. 1988.

[11] C. Connolly, "The determination of next best views," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1985, pp. 432–435.

[12] J. Maver and R. Bajcsy, "Occlusions as a guide for planning the next view," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 5, pp. 417–433, May 1993.

[13] P. Whaite and F. P. Ferrie, "Autonomous exploration: Driven by uncertainty," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 3, pp. 193–205, Mar. 1997.

[14] R. Pito, "A solution to the next best view problem for automated surface acquisition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 10, pp. 1016–1030, Oct. 1999.

[15] R. Zeng, Y. Wen, W. Zhao, and Y.-J. Liu, "View planning in robot active vision: A survey of systems, algorithms, and applications," *Comput. Vis. Media*, vol. 6, pp. 225–245, 2020.

[16] D. Fox, W. Burgard, and S. Thrun, "Active Markov localization for mobile robots," *Robot. Auton. Syst.*, vol. 25, no. 3/4, pp. 195–207, 1998.

[17] G. Borghi and V. Caglioti, "Minimum uncertainty explorations in the self-localization of mobile robots," *IEEE Trans. Robot. Autom.*, vol. 14, no. 6, pp. 902–911, Dec. 1998.

[18] P. Jensfelt and S. Kristensen, "Active global localization for a mobile robot using multiple hypothesis tracking," *IEEE Trans. Robot. Autom.*, vol. 17, no. 5, pp. 748–760, Oct. 2001.

[19] C. Mostegel, A. Wendel, and H. Bischof, "Active monocular localization: Towards autonomous monocular exploration for multirotor MAVs," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 3848–3855.

[20] S. K. Gottipati, K. Seo, D. Bhatt, V. Mai, K. Murthy, and L. Paull, "Deep active localization," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 4394–4401, Oct. 2019.

[21] Q. Xie and Y. Wang, "A survey of filtering based active localization methods," in *Proc. 4th Int. Conf. Big Data Internet Things*, 2020, pp. 69–73.

[22] J. Strader, K. Otsu, and A.-A. Agha-mohammadi, "Perception-aware autonomous mast motion planning for planetary exploration rovers," *J. Field Robot.*, vol. 37, no. 5, pp. 812–829, 2020.

[23] N. Roy, W. Burgard, D. Fox, and S. Thrun, "Coastal navigation-mobile robot navigation with uncertainty in dynamic environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1999, pp. 35–40.

[24] H. Carrillo, I. Reid, and J. A. Castellanos, "On the comparison of uncertainty criteria for active SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 2080–2087.

[25] S. B. Thrun and K. Möller, "Active exploration in dynamic environments," in *Proc. Adv. Neural Inf. Process. Syst.*, 1991, pp. 531–538.

[26] H. J. S. Feder, J. J. Leonard, and C. M. Smith, "Adaptive mobile robot navigation and mapping," *Int. J. Robot. Res.*, vol. 18, no. 7, pp. 650–668, 1999.

[27] F. Bourgault, A. A. Makarenko, S. B. Williams, B. Grocholsky, and H. F. Durrant-Whyte, "Information based adaptive robotic exploration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2002, pp. 540–545.

[28] A. A. Makarenko, S. B. Williams, F. Bourgault, and H. F. Durrant-Whyte, "An experiment in integrated exploration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2002, pp. 534–539.

[29] C. Stachniss, D. Hahnel, and W. Burgard, "Exploration with active loop-closing for FastSLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2004, pp. 1505–1510.

[30] P. Newman, M. Bosse, and J. Leonard, "Autonomous feature-based exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2003, pp. 1234–1240.

[31] C. Stachniss, *Robotic Mapping and Exploration*, vol. 55. Berlin, Germany: Springer, 2009.

[32] R. Platt Jr, R. Tedrake, L. Kaelbling, and T. Lozano-Pérez, "Belief space planning assuming maximum likelihood observations," in *Proc. Robot.: Sci. Syst.*, 2010.

[33] C. Stachniss and W. Burgard, "Mapping and exploration with mobile robots using coverage maps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2003, pp. 467–472.

[34] R. Sim and N. Roy, "Global A-optimal robot exploration in SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 661–666.

[35] A. J. Davison and D. W. Murray, "Simultaneous localization and map-building using active vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 865–880, Jul. 2002.

[36] I. Lluvia, E. Lazkano, and A. Ansuategi, "Active mapping and robot exploration: A survey," *Sensors*, vol. 21, no. 7, 2021, Art. no. 2445.

[37] M. Montemerlo et al., "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proc. 18th AAAI Conf. Artif. Intell.*, 2002, pp. 593–598.

[38] C. Stachniss, G. Grisetti, and W. Burgard, "Information gain-based exploration using Rao-Blackwellized particle filters," in *Proc. Robot.: Sci. Syst.*, 2005, pp. 65–72.

[39] C. Leung, S. Huang, and G. Dissanayake, "Active SLAM using model predictive control and attractor based exploration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2006, pp. 5026–5031.

[40] R. Valencia, J. V. Miro, G. Dissanayake, and J. Andrade-Cetto, "Active pose SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 1885–1891.

[41] V. Ila, J. M. Porta, and J. Andrade-Cetto, "Information-based compact pose SLAM," *IEEE Trans. Robot.*, vol. 26, no. 1, pp. 78–93, Feb. 2010.

[42] L. Carlone, J. Du, M. K. Ng, B. Bona, and M. Indri, "Active SLAM and exploration with particle filters using Kullback-Leibler divergence," *J. Intell. Robot. Syst.*, vol. 75, no. 2, pp. 291–311, 2014.

[43] V. Indelman, L. Carlone, and F. Dellaert, "Planning in the continuous domain: A generalized belief space approach for autonomous navigation in unknown environments," *Int. J. Robot. Res.*, vol. 34, no. 7, pp. 849–882, 2015.

[44] C. Zhu, R. Ding, M. Lin, and Y. Wu, "A 3D frontier-based exploration tool for MAVs," in *Proc. IEEE 27th Int. Conf. Tools Artif. Intell.*, 2015, pp. 348–352.

[45] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-D mapping with an RGB-D camera," *IEEE Trans. Robot.*, vol. 30, no. 1, pp. 177–187, Feb. 2014.

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

[47] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, "Robust visual inertial odometry using a direct EKF-based approach," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 298–304.

[48] C. Papachristos, S. Khattak, and K. Alexis, "Uncertainty-aware receding horizon exploration and mapping using aerial robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 4568–4575.

[49] H. Umari and S. Mukhopadhyay, "Autonomous robotic exploration based on multiple rapidly-exploring randomized trees," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 1396–1402.

[50] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with Rao-Blackwellized particle filters," *IEEE Trans. Robot.*, vol. 23, no. 1, pp. 34–46, Feb. 2007.

[51] H. Carrillo, P. Dames, V. Kumar, and J. A. Castellanos, "Autonomous robotic exploration using a utility function based on Rényi's general theory of entropy," *Auton. Robots*, vol. 42, no. 2, pp. 235–256, 2018.

[52] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental smoothing and mapping," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1365–1378, Dec. 2008.

[53] M. G. Jadidi, J. V. Miro, and G. Dissanayake, "Gaussian processes autonomous mapping and exploration for range-sensing mobile robots," *Auton. Robots*, vol. 42, no. 2, pp. 273–290, 2018.

[54] N. Palomeras, N. Hurtós, E. Vidal, and M. Carreras, "Autonomous exploration of complex underwater environments using a probabilistic next-best-view planner," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1619–1625, Apr. 2019.

[55] G. Grisetti, R. Kümmerle, H. Strasdat, and K. Konolige, "g2o: A general framework for (hyper) graph optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2011, pp. 9–13.

[56] D. S. Chaplot, D. Gandhi, S. Gupta, A. Gupta, and R. Salakhutdinov, "Learning to explore using active neural SLAM," in *Proc. Int. Conf. Learn. Representations*, 2020.

[57] F. Niroui, K. Zhang, Z. Kashino, and G. Nejat, "Deep reinforcement learning robot for search and rescue applications: Exploration in unknown cluttered environments," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 610–617, Apr. 2019.

[58] F. Chen, J. D. Martin, Y. Huang, J. Wang, and B. Englot, "Autonomous exploration under uncertainty via deep reinforcement learning on graphs," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 6140–6147.

[59] H. Li, Q. Zhang, and D. Zhao, "Deep reinforcement learning-based automatic exploration for navigation in unknown environment," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 6, pp. 2064–2076, Jun. 2020.

[60] K. Konolige, G. Grisetti, R. Kümmerle, W. Burgard, B. Limketkai, and R. Vincent, "Efficient sparse pose adjustment for 2D mapping," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 22–29.

[61] S. Suresh, P. Sodhi, J. G. Mangelson, D. Wettergreen, and M. Kaess, "Active SLAM using 3D submap saliency for underwater volumetric exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 3132–3138.

[62] Y. Chen, S. Huang, and R. Fitch, "Active SLAM for mobile robots with area coverage and obstacle avoidance," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 3, pp. 1182–1192, Jun. 2020.

[63] L. Zhao, S. Huang, and G. Dissanayake, "Linear SLAM: Linearising the SLAM problems using submap joining," *Automatica*, vol. 100, pp. 231–246, 2019.

[64] A. Batinovic, T. Petrovic, A. Ivanovic, F. Petric, and S. Bogdan, "A multi-resolution frontier-based planner for autonomous 3D exploration," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4528–4535, Jul. 2021.

[65] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2D LiDAR SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 1271–1278.

[66] J. A. Placed, J. J. G. Rodríguez, J. D. Tardós, and J. A. Castellanos, "ExplORB-SLAM: Active visual SLAM exploiting the pose-graph topology," in *Proc. Iberian Robot. Conf.*, 2023, pp. 199–210.

[67] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017.

[68] E. Bonetto, P. Goldschmid, M. Pabst, M. J. Black, and A. Ahmad, "iRotate: Active visual SLAM for omnidirectional robots," *Robot. Auton. Syst.*, vol. 154, 2022, Art. no. 104102.

[69] M. Labbé and F. Michaud, "RTAB-Map as an open-source LiDAR and visual simultaneous localization and mapping library for large-scale and long-term online operation," *J. Field Robot.*, vol. 36, no. 2, pp. 416–446, 2019.

[70] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artif. Intell.*, vol. 101, no. 1/2, pp. 99–134, 1998.

[71] O. Sigaud and O. Buffet, *Markov Decision Processes in Artificial Intelligence*. Hoboken, NJ, USA: Wiley, 2013.

[72] K. J. Åström, "Optimal control of Markov processes with incomplete state information," *J. Math. Anal. Appl.*, vol. 10, no. 1, pp. 174–205, 1965.

[73] M. Araya, O. Buffet, V. Thomas, and F. Charpillet, "A POMDP extension with belief-dependent rewards," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 64–72.

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

[75] R. Bormann, F. Jordan, W. Li, J. Hampp, and M. Hägele, "Room segmentation: Survey, implementation, and analysis," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 1019–1026.

[76] B. Mu, M. Giamou, L. Paull, A.-A. Agha-mohammadi, J. Leonard, and J. How, "Information-based active SLAM via topological feature graphs," in *Proc. IEEE Conf. Decis. Control*, 2016, pp. 5583–5590.

[77] L. Fermin-Leon, J. Neira, and J. A. Castellanos, "TIGRE: Topological graph based robotic exploration," in *Proc. Eur. Conf. Mobile Robots*, 2017, pp. 1–6.

[78] N. Atanasov, J. Le Ny, K. Daniilidis, and G. J. Pappas, "Decentralized active information acquisition: Theory and application to multi-robot SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 4775–4782.

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

[80] H. P. Moravec, "Sensor fusion in certainty grids for mobile robots," in *Sensor Devices and Systems for Robotics*. Berlin, Germany: Springer, 1989, pp. 253–276.

[81] J. A. Placed and J. A. Castellanos, "Fast autonomous robotic exploration using the underlying graph structure," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 6649–6656.

[82] A. Hornung, K. M. Wurm, C. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Auton. Robots*, vol. 34, no. 3, pp. 189–206, 2013.

[83] E. Vespa, N. Nikolov, M. Grimm, L. Nardi, P. H. Kelly, and S. Leutenegger, "Efficient octree-based volumetric SLAM supporting signed-distance and occupancy mapping," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 1144–1151, Apr. 2018.

[84] M. Muglikar, Z. Zhang, and D. Scaramuzza, "Voxel map for visual SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 4181–4187.

[85] N. Palomeras, M. Carreras, and J. Andrade-Cetto, "Active SLAM for autonomous underwater exploration," *Remote Sens.*, vol. 11, no. 23, 2019, Art. no. 2827.

[86] M. Selin, M. Tiger, D. Duberg, F. Heintz, and P. Jensfelt, "Efficient autonomous exploration planning of large-scale 3D environments," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1699–1706, Apr. 2019.

[87] D. Deng, Z. Xu, W. Zhao, and K. Shimada, "Frontier-based automatic-differentiable information gain measure for robotic exploration of unknown 3D environments," 2020, *arXiv:2011.05288*.

[88] A. Dai, S. Papatheodorou, N. Funk, D. Tzoumanikas, and S. Leutenegger, "Fast frontier-based information-driven autonomous exploration with an MAV," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 9570–9576.

[89] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, "Voxblox: Incremental 3D Euclidean signed distance fields for on-board MAV planning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 1366–1373.

[90] K. Saulnier, N. Atanasov, G. J. Pappas, and V. Kumar, "Information theoretic active exploration in signed distance fields," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 4080–4085.

[91] S. L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas, "Probabilistic data association for semantic SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 1722–1729.

[92] L. Nicholson, M. Milford, and N. Sünderhauf, "Quadricslam: Dual quadrics from object detections as landmarks in object-oriented SLAM," *IEEE Robot. Autom. Lett.*, vol. 4, no. 1, pp. 1–8, Jan. 2019.

[93] R. Eidenberger and J. Scharinger, "Active perception and scene modeling by planning with probabilistic 6D object poses," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 1036–1043.

[94] N. Atanasov, B. Sankaran, J. L. Ny, G. Pappas, and K. Daniilidis, "Nonmyopic view planning for active object classification and pose estimation," *IEEE Trans. Robot.*, vol. 30, no. 5, pp. 1078–1090, Oct. 2014.

[95] M. Grinvald et al., "Volumetric instance-aware semantic mapping and 3D object discovery," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 3037–3044, Jul. 2019.

[96] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, "Kimera: An open-source library for real-time metric-semantic localization and mapping," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 1689–1696.

[97] J. McCormac, R. Clark, M. Bloesch, A. Davison, and S. Leutenegger, "Fusion++: Volumetric object-level SLAM," in *Proc. IEEE Int. Conf. 3D Vis.*, 2018, pp. 32–41.

[98] L. Zheng et al., "Active scene understanding via online semantic reconstruction," *Comput. Graph. Forum*, vol. 38, no. 7, pp. 103–114, 2019.

[99] A. Asgharivaskasi and N. Atanasov, "Active Bayesian multi-class mapping from range and semantic segmentation observations," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 1–7.

[100] A. Asgharivaskasi and N. Atanasov, "Semantic OcTree mapping and Shannon mutual information computation for robot exploration," 2022, *arXiv:2112.04063*.

[101] S. Thrun and A. Bücken, "Integrating grid-based and topological maps for mobile robot navigation," in *Proc. 13th AAAI Conf. Artif. Intell.*, 1996, pp. 944–951.

[102] N. Tomatis, I. Nourbakhsh, and R. Siegwart, "Hybrid simultaneous localization and map building: A natural integration of topological and metric," *Robot. Auton. Syst.*, vol. 44, no. 1, pp. 3–14, 2003.

[103] A. Rosinol et al., "Kimera: From SLAM to spatial perception with 3D dynamic scene graphs," *Int. J. Robot. Res.*, vol. 40, no. 12–14, pp. 1510–1546, 2021.

[104] C. Gomez, A. C. Hernandez, and R. Barber, "Topological frontier-based exploration and map-building using semantic information," *Sensors*, vol. 19, no. 20, 2019, Art. no. 4595.

[105] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, "Coordinated multi-robot exploration," *IEEE Trans. Robot.*, vol. 21, no. 3, pp. 376–386, Jun. 2005.

[106] H. H. González-Banos and J.-C. Latombe, "Navigation strategies for exploring indoor environments," *Int. J. Robot. Res.*, vol. 21, no. 10/11, pp. 829–848, 2002.

[107] B. Tovar, L. Munoz-Gómez, R. Murrieta-Cid, M. Alencastre-Miranda, R. Monroy, and S. Hutchinson, "Planning exploration strategies for simultaneous localization and mapping," *Robot. Auton. Syst.*, vol. 54, no. 4, pp. 314–331, 2006.

[108] P. Quin, D. D. K. Nguyen, T. L. Vu, A. Alempijevic, and G. Paul, "Approaches for efficiently detecting frontier cells in robotics exploration," *Front. Robot. AI*, vol. 8, 2021, Art. no. 616470.

[109] M. Keidar and G. A. Kaminka, "Robot exploration with fast frontier detection: Theory and experiments," in *Proc. 11th Int. Conf. Auton. Agents Multiagent Syst.*, 2012, pp. 113–120.

[110] M. Keidar and G. A. Kaminka, "Efficient frontier detection for robot exploration," *Int. J. Robot. Res.*, vol. 33, no. 2, pp. 215–236, 2014.

[111] D. Holz, N. Basilico, F. Amigoni, and S. Behnke, "Evaluating the efficiency of frontier-based exploration strategies," in *Proc. IEEE 41st Int. Symp. Robot.*, 2010, pp. 1–8.

[112] C.-Y. Wu and H.-Y. Lin, "Autonomous mobile robot exploration in unknown indoor environments based on rapidly-exploring random tree," in *Proc. IEEE Int. Conf. Ind. Technol.*, 2019, pp. 1345–1350.

[113] W. Qiao, Z. Fang, and B. Si, "Sample-based frontier detection for autonomous robot exploration," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2018, pp. 1165–1170.

[114] C. Dornhege and A. Kleiner, "A frontier-void-based approach for autonomous exploration in 3D," *Adv. Robot.*, vol. 27, no. 6, pp. 459–468, 2013.

[115] P. Senarathne and D. Wang, "Towards autonomous 3D exploration using surface frontiers," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot.*, 2016, pp. 34–41.

[116] S. Shen, N. Michael, and V. Kumar, "Stochastic differential equation-based exploration algorithm for autonomous indoor 3D exploration with a micro-aerial vehicle," *Int. J. Robot. Res.*, vol. 31, no. 12, pp. 1431–1444, 2012.

[117] L. Lu, C. Redondo, and P. Campoy, "Optimal frontier-based autonomous exploration in unconstructed environment using RGB-D sensor," *Sensors*, vol. 20, no. 22, 2020, Art. no. 6507.

[118] R. Grabowski, P. Khosla, and H. Choset, "Autonomous exploration via regions of interest," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2003, pp. 1691–1696.

[119] A. Kim and R. M. Eustice, "Active visual SLAM for robotic area coverage: Theory and experiment," *Int. J. Robot. Res.*, vol. 34, no. 4/5, pp. 457–475, 2015.

[120] J. van den Berg, S. Patil, and R. Alterovitz, "Motion planning under uncertainty using iterative local optimization in belief space," *Int. J. Robot. Res.*, vol. 31, no. 11, pp. 1263–1278, 2012.

[121] L. Tai, G. Paolo, and M. Liu, "Virtual-to-real deep reinforcement learning: Continuous control of mobile robots for mapless navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 31–36.

[122] D. S. Chaplot, E. Parisotto, and R. Salakhutdinov, "Active neural localization," in *Proc. Int. Conf. Learn. Representations*, 2018.

[123] J. A. Placed and J. A. Castellanos, "A deep reinforcement learning approach for active SLAM," *Appl. Sci.*, vol. 10, no. 23, 2020, Art. no. 8386.

[124] B. Charrow et al., "Information-theoretic planning with trajectory optimization for dense 3D mapping," in *Proc. Robot.: Sci. Syst.*, 2015, pp. 3–12.

[125] F. Amigoni and A. Gallo, "A multi-objective exploration strategy for mobile robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 3850–3855.

[126] W. Chen and L. Liu, "Pareto Monte Carlo tree search for multi-objective informative planning," in *Proc. Robot.: Sci. Syst.*, 2019.

[127] A. Soragna, M. Baldini, D. Joho, R. Kümmerle, and G. Grisetti, "Active slam using connectivity graphs as priors," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 340–346.

[128] Y. Chen, L. Zhao, K. M. B. Lee, C. Yoo, S. Huang, and R. Fitch, "Broadcast your weaknesses: Cooperative active pose-graph SLAM for multiple robots," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2200–2207, Apr. 2020.

[129] M. Juliá, A. Gil, and O. Reinoso, "A comparison of path planning strategies for autonomous exploration and mapping of unknown environments," *Auton. Robots*, vol. 33, no. 4, pp. 427–444, 2012.

[130] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, 1948.

[131] T. Cover, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 1999.

[132] R. G. Colares and L. Chaimowicz, "The next frontier: Combining information gain and distance cost for decentralized multi-robot exploration," in *Proc. ACM Symp. Appl. Comput.*, 2016, pp. 268–274.

[133] J. Vallvé and J. Andrade-Cetto, "Dense entropy decrease estimation for mobile robot exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 6083–6089.

[134] J.-L. Blanco, J.-A. Fernandez-Madrigal, and J. González, "A novel measure of uncertainty for mobile robot SLAM with Rao-Blackwellized particle filters," *Int. J. Robot. Res.*, vol. 27, no. 1, pp. 73–89, 2008.

[135] 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. IEEE Int. Conf. Robot. Autom.*, 2020, pp. 10751–10757.

[136] J. Wang and B. Englot, "Autonomous exploration with expectation-maximization," in *Robotics Research*. Berlin, Germany: Springer, 2020, pp. 759–774.

[137] A. J. Davison, "Active search for real-time vision," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005, pp. 66–73.

[138] J. Du, L. Carlone, M. K. Ng, B. Bona, and M. Indri, "A comparative study on active SLAM and autonomous exploration with particle filters," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, 2011, pp. 916–923.

[139] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.

[140] L. Mihaylova, T. Lefebvre, H. Bruyninckx, K. Gadeyne, and J. De Schutter, "A comparison of decision making criteria and optimization methods for active robotic sensing," in *Proc. Int. Conf. Numer. Methods Appl.*, 2002, pp. 316–324.

[141] R. Houthooft, X. Chen, Y. Duan, J. Schulman, F. De Turck, and P. Abbeel, "VIME: Variational information maximizing exploration," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1117–1125.

[142] D. Fox, "Adapting the sample size in particle filters through KLD-sampling," *Int. J. Robot. Res.*, vol. 22, no. 12, pp. 985–1003, 2003.

[143] M. Kontitsis, E. A. Theodorou, and E. Todorov, "Multi-robot active SLAM with relative entropy optimization," in *Proc. Amer. Control Conf.*, 2013, pp. 2757–2764.

[144] D. Deng, R. Duan, J. Liu, K. Sheng, and K. Shimada, "Robotic exploration of unknown 2D environment using a frontier-based automatic-differentiable information gain measure," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics*, 2020, pp. 1497–1503.

[145] D. S. Levine and J. P. How, "Sensor selection in high-dimensional Gaussian trees with nuisances," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2211–2219.

[146] D. Kopitkov and V. Indelman, "No belief propagation required: Belief space planning in high-dimensional state spaces via factor graphs, the matrix determinant lemma, and re-use of calculation," *Int. J. Robot. Res.*, vol. 36, no. 10, pp. 1088–1130, 2017.

[147] D. Kopitkov and V. Indelman, "General-purpose incremental covariance update and efficient belief space planning via a factor-graph propagation action tree," *Int. J. Robot. Res.*, vol. 38, no. 14, pp. 1644–1673, 2019.

[148] H. Chernoff, "Locally optimal designs for estimating parameters," *Ann. Math. Statist.*, vol. 24, pp. 586–602, 1953.

[149] S. Ehrenfeld, "On the efficiency of experimental designs," *Ann. Math. Statist.*, vol. 26, no. 2, pp. 247–255, 1955.

[150] A. Wald, "On the efficient design of statistical investigations," *Ann. Math. Statist.*, vol. 14, no. 2, pp. 134–140, 1943.

[151] J. Kiefer, "General equivalence theory for optimum designs (approximate theory)," *Ann. Statist.*, vol. 2, pp. 849–879, 1974.

[152] L. Carlone and S. Karaman, "Attention and anticipation in fast visual-inertial navigation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 3886–3893.

[153] H. Carrillo, Y. Latif, M. L. Rodriguez-Arevalo, J. Neira, and J. A. Castellanos, "On the monotonicity of optimality criteria during exploration in active SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 1476–1483.

[154] Y. Kim and A. Kim, "On the uncertainty propagation: Why uncertainty on lie groups preserves monotonicity?," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 3425–3432.

[155] M. L. Rodríguez-Arévalo, J. Neira, and J. A. Castellanos, "On the importance of uncertainty representation in active SLAM," *IEEE Trans. Robot.*, vol. 34, no. 3, pp. 829–834, Jun. 2018.

[156] C.-S. Cheng, "Maximizing the total number of spanning trees in a graph: Two related problems in graph theory and optimum design theory," *J. Combinatorial Theory, Ser. B*, vol. 31, no. 2, pp. 240–248, 1981.

[157] K. Khosoussi, S. Huang, and G. Dissanayake, "Novel insights into the impact of graph structure on SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 2707–2714.

[158] K. Khosoussi, M. Giamou, G. S. Sukhatme, S. Huang, G. Dissanayake, and J. P. How, "Reliable graphs for SLAM," *Int. J. Robot. Res.*, vol. 38, no. 2/3, pp. 260–298, 2019.

[159] Y. Chen, S. Huang, L. Zhao, and G. Dissanayake, "Cramér–Rao bounds and optimal design metrics for pose-graph SLAM," *IEEE Trans. Robot.*, vol. 37, no. 2, pp. 627–641, Apr. 2021.

[160] J. A. Placed and J. A. Castellanos, "A general relationship between optimality criteria and connectivity indices for active graph-SLAM," *IEEE Robot. Autom. Lett.*, vol. 8, no. 2, pp. 816–823, Feb. 2022.

[161] J. A. Placed and J. A. Castellanos, "Enough is enough: Towards autonomous uncertainty-driven stopping criteria," *IFAC-PapersOnLine*, vol. 55, no. 14, pp. 126–132, 2022.

[162] A. Kitanov and V. Indelman, "Topological information-theoretic belief space planning with optimality guarantees," 2019, *arXiv:1903.00927*.

[163] M. Shienman, A. Kitanov, and V. Indelman, "FT-BSP: Focused topological belief space planning," *IEEE Robot. Autom. Lett.*, vol. 6, no. 3, pp. 4744–4751, Jul. 2021.

[164] S. M. LaValle and J. J. Kuffner Jr, "Randomized kinodynamic planning," *Int. J. Robot. Res.*, vol. 20, no. 5, pp. 378–400, 2001.

[165] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. Robot. Autom.*, vol. 12, no. 4, pp. 566–580, Aug. 1996.

[166] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 846–894, 2011.

[167] G. Oriolo, M. Vendittelli, L. Freda, and G. Troso, "The SRT method: Randomized strategies for exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2004, pp. 4688–4694.

[168] L. Freda and G. Oriolo, "Frontier-based probabilistic strategies for sensor-based exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 3881–3887.

[169] E. Bonetto, P. Goldschmid, M. J. Black, and A. Ahmad, "Active visual SLAM with independently rotating camera," in *Proc. Eur. Conf. Mobile Robots*, 2021, pp. 1–8.

[170] J. Van den Berg, S. Patil, and R. Alterovitz, "Efficient approximate value iteration for continuous Gaussian POMDPs," in *Proc. 26th AAAI Conf. Artif. Intell.*, 2012, pp. 1832–1838.

[171] O. Madani, S. Hanks, and A. Condon, "On the undecidability of probabilistic planning and infinite-horizon partially observable Markov decision problems," in *Proc. 16th AAAI Conf. Artif. Intell.*, 1999, pp. 541–548.

[172] H. Bai, D. Hsu, and W. S. Lee, "Integrated perception and planning in the continuous space: A POMDP approach," *Int. J. Robot. Res.*, vol. 33, no. 9, pp. 1288–1302, 2014.

[173] J. M. Porta, M. T. Spaan, and N. Vlassis, "Robot planning in partially observable continuous domains," in *Proc. Robot.: Sci. Syst.*, 2005, pp. 217–224.

[174] S. Prentice and N. Roy, "The belief roadmap: Efficient planning in belief space by factoring the covariance," *Int. J. Robot. Res.*, vol. 28, no. 11/12, pp. 1448–1465, 2009.

[175] R. Valencia, M. Morta, J. Andrade-Cetto, and J. M. Porta, "Planning reliable paths with pose SLAM," *IEEE Trans. Robot.*, vol. 29, no. 4, pp. 1050–1059, Aug. 2013.

[176] J. L. Ny and G. Pappas, "On trajectory optimization for active sensing in Gaussian process models," in *Proc. IEEE Conf. Decis. Control*, 2009, pp. 6286–6292.

[177] S. Koga, A. Asgharivaskasi, and N. Atanasov, "Active exploration and mapping via iterative covariance regulation over continuous $SE(3)$ trajectories," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 2735–2741.

[178] B. Schlotfeldt, N. Atanasov, and G. J. Pappas, "Maximum information bounds for planning active sensing trajectories," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 4913–4920.

[179] G. Hollinger and G. Sukhatme, "Sampling-based robotic information gathering algorithms," *Int. J. Robot. Res.*, vol. 33, no. 9, pp. 1271–1287, 2014.

[180] X. Lan and M. Schwager, "Rapidly-exploring random cycles: Persistent estimation of spatio-temporal fields with multiple sensing robots," *IEEE Trans. Robot.*, vol. 32, no. 5, pp. 1230–1244, Oct. 2016.

[181] Y. Kantaros, B. Schlotfeldt, N. Atanasov, and G. J. Pappas, "Asymptotically optimal planning for non-myopic multi-robot information gathering," in *Proc. Robot.: Sci. Syst.*, 2019, pp. 22–26.

[182] S. Koga, A. Asgharivaskasi, and N. Atanasov, "Active SLAM over continuous trajectory and control: A covariance-feedback approach," in *Proc. Amer. Control Conf.*, 2022, pp. 5062–5068.

[183] S. Rahman and S. L. Waslander, "Uncertainty-constrained differential dynamic programming in belief space for vision based robots," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3112–3119, Apr. 2021.

[184] V. Murali, I. Spasojevic, W. Guerra, and S. Karaman, "Perception-aware trajectory generation for aggressive quadrotor flight using differential flatness," in *Proc. Amer. Control Conf.*, 2019, pp. 3936–3943.

[185] L. Carlone and D. Lyons, "Uncertainty-constrained robot exploration: A mixed-integer linear programming approach," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2014, pp. 1140–1147.

[186] C. Leung, S. Huang, and G. Dissanayake, "Active SLAM in structured environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 1898–1903.

[187] M. Xu, Y. Song, Y. Chen, S. Huang, and Q. Hao, "Invariant EKF based 2D active SLAM with exploration task," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 5350–5356.

[188] R. Martinez-Cantin, N. De Freitas, E. Brochu, J. Castellanos, and A. Doucet, "A bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot," *Auton. Robots*, vol. 27, no. 2, pp. 93–103, 2009.

[189] P. Karkus, D. Hsu, and W. S. Lee, "QMDP-Net: Deep learning for planning under partial observability," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4697–4707.

[190] S. Bai, F. Chen, and B. Englot, "Toward autonomous mapping and exploration for mobile robots through deep supervised learning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 2379–2384.

[191] F. Chen, J. Wang, T. Shan, and B. Englot, "Autonomous exploration under uncertainty via graph convolutional networks," in *Proc. Int. Symp. Robot. Res.*, 2019, pp. 676–691.

[192] V. Mnih et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[193] H. V. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 2094–2100.

[194] M. Hessel et al., "Rainbow: Combining improvements in deep reinforcement learning," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 3215–3222.

[195] T. P. Lillicrap et al., "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*.

[196] V. Mnih et al., "Asynchronous methods for deep reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1928–1937.

[197] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep reinforcement learning: A brief survey," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 26–38, Nov. 2017.

[198] F. Zeng, C. Wang, and S. S. Ge, "A survey on visual navigation for artificial agents with deep reinforcement learning," *IEEE Access*, vol. 8, pp. 135426–135442, 2020.

[199] L. Tai and M. Liu, "Mobile robots exploration through CNN-based reinforcement learning," *Robot. Biomimetics*, vol. 3, no. 1, pp. 1–8, 2016.

[200] P. Mirowski et al., "Learning to navigate in complex environments," in *Proc. Int. Conf. Learn. Representations*, 2017.

[201] R. M. Ryan and E. L. Deci, "Intrinsic and extrinsic motivations: Classic definitions and new directions," *Contemporary Educ. Psychol.*, vol. 25, no. 1, pp. 54–67, 2000.

[202] M. Bellemare, S. Srinivasan, G. Ostrovski, T. Schaul, D. Saxton, and R. Munos, "Unifying count-based exploration and intrinsic motivation," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 1471–1479.

[203] T. Chen, S. Gupta, and A. Gupta, "Learning exploration policies for navigation," in *Proc. Int. Conf. Learn. Representations*, 2019.

[204] O. Zhelo, J. Zhang, L. Tai, M. Liu, and W. Burgard, "Curiosity-driven exploration for mapless navigation with deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Autom. Workshop Mach. Learn. Plan. Control Robot Motion*, 2018.

[205] J. Hu, H. Niu, J. Carrasco, B. Lennox, and F. Arvin, "Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 14413–14423, Dec. 2020.

[206] 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. IEEE Int. Conf. Robot. Autom.*, 2022, pp. 4466–4472.

[207] K. Zhu and T. Zhang, "Deep reinforcement learning based mobile robot navigation: A review," *Tsinghua Sci. Technol.*, vol. 26, no. 5, pp. 674–691, 2021.

[208] M. Hausknecht and P. Stone, "Deep recurrent Q-learning for partially observable MDPs," in *Proc. AAAI Fall Symp. Ser.*, 2015.

[209] H. Hu, K. Zhang, A. H. Tan, M. Ruan, C. Agia, and G. Nejat, "A sim-to-real pipeline for deep reinforcement learning for autonomous robot navigation in cluttered rough terrain," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 6569–6576, Oct. 2021.

[210] K. Yokoyama and K. Morioka, "Autonomous mobile robot with simple navigation system based on deep reinforcement learning and a monocular camera," in *Proc. IEEE/SICE Int. Symp. Syst. Integration*, 2020, pp. 525–530.

[211] H. Shi, L. Shi, M. Xu, and K.-S. Hwang, "End-to-end navigation strategy with deep reinforcement learning for mobile robots," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2393–2402, Apr. 2020.

[212] B. Gerkey, R. T. Vaughan, and A. Howard, "The player/stage project: Tools for multi-robot and distributed sensor systems," in *Proc. 11th Int. Conf. Adv. Robot.*, 2003, pp. 317–323.

[213] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, An open-source multi-robot simulator," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2004, pp. 2149–2154.

[214] E. Rohmer, S. P. Singh, and M. Freese, "V-REP: A versatile and scalable robot simulation framework," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2013, pp. 1321–1326.

[215] I. Zamora, N. G. Lopez, V. M. Vilches, and A. H. Cordero, "Extending the OpenAI gym for robotics: A toolkit for reinforcement learning using ROS and Gazebo," 2016, *arXiv:1608.05742*.

[216] G. Brockman et al., "OpenAI gym," 2016, *arXiv:1606.01540*.

[217] C. Beattie et al., "Deepmind lab," 2016, *arXiv:1612.03801*.

[218] M. Savva et al., "Habitat: A platform for embodied AI research," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9339–9347.

[219] J. Straub et al., "The Replica dataset: A digital replica of indoor spaces," 2019, *arXiv:1906.05797*.

[220] C. Li et al., "iGibson 2.0: Object-centric simulation for robot learning of everyday household tasks," in *Proc. 5th Conf. Robot Learn.*, 2022, pp. 455–465.

[221] B. Charrow, V. Kumar, and N. Michael, "Approximate representations for multi-robot control policies that maximize mutual information," *Auton. Robots*, vol. 37, pp. 383–400, 2014.

[222] Y. Kantaros, B. Schlotfeldt, N. Atanasov, and G. J. Pappas, "Sampling-based planning for non-myopic multi-robot information gathering," *Auton. Robots*, vol. 45, pp. 1029–1046, 2021.

[223] W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2000, pp. 476–481.

[224] B. Schlotfeldt, D. Thakur, N. Atanasov, V. Kumar, and G. J. Pappas, "Anytime planning for decentralized multi-robot active information gathering," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 1025–1032, Apr. 2018.

[225] M. Ossenkopf, G. Castro, F. Pessacg, K. Geihs, and P. De Cristóforis, "Long-horizon active SLAM system for multi-agent coordinated exploration," in *Proc. Eur. Conf. Mobile Robots*, 2019, pp. 1–6.

[226] M. Lauri, E. Heinänen, and S. Frintrop, "Multi-robot active information gathering with periodic communication," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2017, pp. 851–856.

[227] Z. I. Botev, D. P. Kroese, R. Y. Rubinstein, and P. L'Ecuyer, "The cross-entropy method for optimization," in *Handbook of Statistics*, vol. 31. Amsterdam, The Netherlands: Elsevier, 2013, pp. 35–59.

[228] V. Indelman, "Cooperative multi-robot belief space planning for autonomous navigation in unknown environments," *Auton. Robots*, vol. 42, no. 2, pp. 353–373, 2018.

[229] T. Regev and V. Indelman, "Multi-robot decentralized belief space planning in unknown environments via efficient re-evaluation of impacted paths," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 5591–5598.

[230] G. Best, J. Faigl, and R. Fitch, "Multi-robot path planning for budgeted active perception with self-organising maps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 3164–3171.

[231] M. Fehr, T. Taubner, Y. Liu, R. Siegwart, and C. Cadena, "Predicting unobserved space for planning via depth map augmentation," in *Proc. 19th Int. Conf. Adv. Robot.*, 2019, pp. 30–36.

[232] S. K. Ramakrishnan, Z. Al-Halah, and K. Grauman, "Occupancy anticipation for efficient exploration and navigation," in *Proc. 16th Eur. Conf. Comput. Vis.*, 2020, pp. 400–418.

[233] K. D. Katyal, A. Polevoy, J. Moore, C. Knuth, and K. M. Popek, "High-speed robot navigation using predicted occupancy maps," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 5476–5482.

[234] S. Y. Hayoun, E. Zwecher, E. Iceland, A. Revivo, S. R. Levy, and A. Barel, "Integrating deep-learning-based image completion and motion planning to expedite indoor mapping," 2020, *arXiv:2011.02043*.

[235] R. Shrestha, F.-P. Tian, W. Feng, P. Tan, and R. Vaughan, "Learned map prediction for enhanced mobile robot exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 1197–1204.

[236] A. Dai, D. Ritchie, M. Bokeloh, S. Reed, J. Sturm, and M. Nießner, "ScanComplete: Large-scale scene completion and semantic segmentation for 3D scans," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4578–4587.

[237] C. Richter and N. Roy, "Safe visual navigation via deep learning and novelty detection," in *Proc. Robot.: Sci. Syst.*, 2017.

[238] C. Richter, W. Vega-Brown, and N. Roy, "Bayesian learning for safe high-speed navigation in unknown environments," in *Robotics Research*. Berlin, Germany: Springer, 2018, pp. 325–341.

[239] O. Asraf and V. Indelman, "Experience-based prediction of unknown environments for enhanced belief space planning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 6781–6788.

[240] N. Fairfield and D. Wettergreen, "Active SLAM and loop prediction with the segmented map using simplified models," in *Field and Service Robotics*. Berlin, Germany: Springer, 2010, pp. 173–182.

[241] I. Armeni et al., "3D scene graph: A structure for unified semantics, 3D space, and camera," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 5664–5673.

[242] N. Hughes, Y. Chang, and L. Carlone, "Hydra: A real-time spatial perception system for 3D scene graph construction and optimization," in *Proc. Robot.: Sci. Syst.*, 2022.

[243] Z. Ravichandran, L. Peng, N. Hughes, J. Griffith, and L. Carlone, "Hierarchical representations and explicit memory: Learning effective navigation policies on 3D scene graphs using graph neural networks," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2022, pp. 9272–9279.

[244] N. Sunderhauf and P. Protzel, "Towards a robust back-end for pose graph SLAM," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 1254–1261.

[245] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," *Int. J. Robot. Res.*, vol. 32, no. 7, pp. 826–840, 2013.

[246] H. Yang, P. Antonante, V. Tzoumas, and L. Carlone, "Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1127–1134, Apr. 2020.

[247] V. Indelman, E. Nelson, J. Dong, N. Michael, and F. Dellaert, "Incremental distributed inference from arbitrary poses and unknown data association: Using collaborating robots to establish a common reference," *IEEE Control Syst. Mag.*, vol. 36, no. 2, pp. 41–74, Apr. 2016.

[248] M. Hsiao and M. Kaess, "MH-iSAM2: Multi-hypothesis iSAM using Bayes tree and Hypo-tree," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2019, pp. 1274–1280.

[249] O. Shelly and V. Indelman, "Hypotheses disambiguation in retrospective," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2321–2328, Apr. 2022.

[250] S. Pathak, A. Thomas, and V. Indelman, "A unified framework for data association aware robust belief space planning and perception," *Int. J. Robot. Res.*, vol. 32, no. 2/3, pp. 287–315, 2018.

[251] M. Hsiao, J. G. Mangelson, S. Suresh, C. Debrunner, and M. Kaess, "ARAS: Ambiguity-aware robust active SLAM based on multi-hypothesis state and map estimations," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2020, pp. 5037–5044.

[252] P. M. Dames, "Distributed multi-target search and tracking using the PHD filter," *Auton. Robots*, vol. 44, pp. 673–689, 2020.

[253] M. Shienman and V. Indelman, "D2A-BSP: Distilled data association belief space planning with performance guarantees under budget constraints," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2022, pp. 11058–11065.

[254] V. Indelman, "No correlations involved: Decision making under uncertainty in a conservative sparse information space," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 407–414, Jan. 2016.

[255] K. Elimelech and V. Indelman, "Simplified decision making in the belief space using belief sparsification," *Int. J. Robot. Res.*, vol. 41, no. 5, pp. 470–496, 2022.

[256] M. Barenboim and V. Indelman, "Adaptive information belief space planning," in *Proc. 31st Int. Joint Conf. Artif. Intell. 25th Eur. Conf. Artif. Intell.*, 2022.

[257] J. P. van den Berg and M. H. Overmars, "Roadmap-based motion planning in dynamic environments," *IEEE Trans. Robot.*, vol. 21, no. 5, pp. 885–897, Oct. 2005.

[258] M. R. U. Saputra, A. Markham, and N. Trigoni, "Visual SLAM and structure from motion in dynamic environments: A survey," *ACM Comput. Surv.*, vol. 51, no. 2, pp. 1–36, 2018.

[259] D. Trivun, E. Šalaka, D. Osmankovic, J. Velagić, and N. Osmić, "Active SLAM-based algorithm for autonomous exploration with mobile robot," in *Proc. IEEE Int. Conf. Ind. Technol.*, 2015, pp. 74–79.

[260] I. Maurović, M. Seder, K. Lenac, and I. Petrović, "Path planning for active SLAM based on the D* algorithm with negative edge weights," *IEEE Trans. Syst., Man, Cybern.: Syst.*, vol. 48, no. 8, pp. 1321–1331, Aug. 2018.

[261] E. Anshelevich, S. Owens, F. Lamiraux, and L. E. Kavraki, "Deformable volumes in path planning applications," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2000, pp. 2290–2295.

[262] S. Rodriguez, J.-M. Lien, and N. M. Amato, "Planning motion in completely deformable environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 2466–2471.

[263] R. A. Newcombe, D. Fox, and S. M. Seitz, "DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 343–352.

[264] J. Lamarca, S. Parashar, A. Bartoli, and J. Montiel, "DefSLAM: Tracking and mapping of deforming scenes from monocular sequences," *IEEE Trans. Robot.*, vol. 37, no. 1, pp. 291–303, Feb. 2021.

[265] Y. Xu et al., "Explore-bench: Data sets, metrics and evaluations for frontier-based and deep-reinforcement-learning-based autonomous exploration," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2022, pp. 6225–6231.

[266] A. P. del Pobil, R. Madhavan, and E. Messina, "Benchmarks in robotics research," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. Workshop*, 2006.

[267] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: Using the Yale-CMU-Berkeley object and model set," *IEEE Robot. Autom. Mag.*, vol. 22, no. 3, pp. 36–52, Sep. 2015.

[268] J. Pineau et al., "Improving reproducibility in machine learning research: A report from the NeurIPS 2019 reproducibility program," *J. Mach. Learn. Res.*, vol. 22, pp. 7459–7478, 2021.

[269] P. Ammirato, A. C. Berg, and J. Kosecka, "Active vision dataset benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2018, pp. 2046–2049.

[270] D. Hall et al., "BenchBot environments for active robotics (BEAR): Simulated data for active scene understanding research," *Int. J. Robot. Res.*, 2022, vol. 41, pp. 259–269.

[271] M. Walter, F. Hover, and J. Leonard, "SLAM for ship hull inspection using exactly sparse extended information filters," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 1463–1470.

[272] J. Serafin, M. Di Cicco, T. Bonanni, C. Stachniss, and V. Ziparo, "Robots for exploration, digital preservation and visualization of archaeological sites," in *Artificial Intelligence for Cultural Heritage*. Newcastle Upon Tyne, U.K.: Cambridge Scholars Publishing, 2016, pp. 121–140.

[273] P. Li, C.-Y. Yang, R. Wang, and S. Wang, "A high-efficiency, information-based exploration path planning method for active simultaneous localization and mapping," *Int. J. Adv. Robot. Syst.*, vol. 17, no. 1, 2020, Art. no. 1729881420903207.

Julio A. Placed (Student Member, IEEE) received the B.S. and M.S. degrees in industrial engineering (mention in robotics and industrial automation) from the University of Zaragoza, Zaragoza, Spain, in 2016 and 2019, respectively, where he is currently working toward the Ph.D. degree in systems engineering and computer science with the Robotics, Perception and Real Time Group, Aragón Institute of Engineering Research (I3A).

His research is focused on the active SLAM problem, with efficient, fast and reliable quantification of utility, and the use of novel learning and graph-theoretic techniques being some of his research interests. He is also interested in providing a unified vision of the problem to facilitate its understanding and avoid its dilution in different areas.

Jared Strader received the B.Sc. degree in mathematics from Evangel University, Springfield, MO, USA, in 2012 and the Ph.D. degree in aerospace engineering from West Virginia University, Morgantown, WV, USA, in 2021.

He was a Visiting Researcher with the NASA Jet Propulsion Laboratory, Pasadena, CA, USA, in 2018 and he is currently a Postdoctoral Associate with the Laboratory for Information and Decision Systems (LIDS), Massachusetts Institute of Technology, Cambridge, MA, USA. His research interests include perception and decision making for robotic systems with a focus on active localization in perceptually degraded environments and high-level scene understanding.

Henry Carrillo received the B.Eng. degree in electronics engineering from Universidad del Norte, Barranquilla, Colombia, in 2007, the M.Sc. degree in electronics engineering from Pontificia Universidad Javeriana, Bogotá, Colombia, in 2010, and the M.Sc. and Ph.D. degrees in computer science from Universidad de Zaragoza, Zaragoza, Spain, in 2010 and 2014, respectively. His doctoral thesis was titled "Active SLAM: utility functions and applications" which focuses on utility functions for active SLAM algorithms, in particular delving into their components related to quantifying the uncertainty of the robot and the map.

He is currently a Computer Vision Delivery Lead with Genius Sports, a U.K.-based sports technology company with an office in Medellín, Colombia. Prior to moving to the industry sector in 2019, he was an Associate Professor with Universidad Sergio Arboleda, Bogotá, Colombia, for four years. His research addresses the problem of active perception for autonomous systems.

Nikolay Atanasov (Member, IEEE) received the B.S. degree in electrical engineering from Trinity College, Hartford, CT, USA, in 2008 and the M.S. and Ph.D. degrees in electrical and systems engineering from the University of Pennsylvania, Philadelphia, PA, USA, in 2012 and 2015, respectively.

He is currently an Assistant Professor of Electrical and Computer Engineering with the University of California San Diego, La Jolla, CA, USA. He works on probabilistic models that unify geometric and semantic information in simultaneous localization and mapping (SLAM) and on optimal control and reinforcement learning algorithms for minimizing uncertainty in probabilistic models. His research focuses on robotics, control theory, and machine learning with applications to active perception problems for autonomous mobile robots.

Dr. Atanasov was the recipient of the Joseph and Rosaline Wolf award for the best Ph.D. dissertation in Electrical and Systems Engineering, University of Pennsylvania in 2015, the best conference paper award at the IEEE International Conference on Robotics and Automation (ICRA) in 2017, and the NSF CAREER award in 2021.

Vadim Indelman (Member, IEEE) received the Ph.D. degree in aerospace engineering and the B.A. and B.Sc. degrees in computer science and aerospace engineering from the Technion—Israel Institute of Technology (Technion), Haifa, Israel, in 2002 and 2011, respectively.

He is currently an Associate Professor with the Department of Aerospace Engineering, Technion and he is also a member of the Technion Autonomous Systems Program (TASP), the Technion Artificial Intelligence Hub (Tech.AI), and the Israeli Smart Transportation Research Center (ISTRC). Prior to joining the Technion as a faculty member, he was a Postdoctoral Fellow with the Institute of Robotics and Intelligent Machines (IRIM), Georgia Institute of Technology, Atlanta, GA, USA (between 2012 and 2014). His current research interests include planning under uncertainty, SLAM, and robust and semantic perception in single and multirobot systems.

Luca Carlone (Senior Member, IEEE) received the B.S. and S.M. degrees in mechatronics from the Polytechnic University of Turin, Turin, Italy, in 2006 and 2008, respectively, the S.M. degree in automation engineering from the Polytechnic University of Milan, Milan, Italy, in 2008, and the Ph.D. degree in robotics from the Polytechnic University of Turin, in 2012.

He is currently the Leonardo Career Development Associate Professor with the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, USA, and a Principal Investigator with the Laboratory for Information and Decision Systems (LIDS). He joined LIDS as a Postdoctoral Associate (2015) and later as a Research Scientist (2016), after spending two years as a Postdoctoral Fellow with the Georgia Institute of Technology, Atlanta, GA, USA (2013–2015). His research interests include nonlinear estimation, numerical and distributed optimization, and probabilistic inference, applied to sensing, perception, and decision-making in single and multirobot systems. His work includes seminal results on certifiably correct algorithms for localization and mapping, as well as approaches for visual-inertial navigation and distributed mapping.

Dr. Carlone is a recipient of the Best Student Paper Award at IEEE RSJ International Conference on Intelligent Robots and Systems 2021, the Best Paper Award in Robot Vision at International Conference on Robotics and Automation 2020, a 2020 Honorable Mention from the IEEE ROBOTICS AND AUTOMATION LETTERS, a Track Best Paper award at the 2021 IEEE Aerospace Conference, the 2017 Transactions on Robotics King-Sun Fu Memorial Best Paper Award, the Best Paper Award at Workshop on the Algorithmic Foundations of Robotics 2016, the Best Student Paper Award at the 2018 Symposium on VLSI Circuits, and he was best paper finalist at Robotics Science and Systems (RSS) 2015, RSS 2021, and IEEE/CVF Winter Conference on Applications of Computer Vision 2023. He is also a recipient of the AIAA Aeronautics and Astronautics Advising Award (2022), the NSF CAREER Award (2021), the RSS Early Career Award (2020), the Google Daydream Award (2019), the Amazon Research Award (2020, 2022), and the MIT AeroAstro Vickie Kerrebrock Faculty Award (2020). He is an AIAA Associate Fellow.

José A. Castellanos (Senior Member, IEEE) received the M.S. and Ph.D. degrees in industrial electrical engineering from the University of Zaragoza, Zaragoza, Spain, in 1994 and 1998, respectively.

He is currently a Professor in Systems Engineering and Automation with the University of Zaragoza. He is in charge of courses in SLAM and Control Engineering with the Department of Computer Science and Systems Engineering. His current research interests include SLAM, autonomous vehicle navigation, and decision-making under uncertainty.

Dr. Castellanos was an Associate Editor for the IEEE TRANSACTIONS ON ROBOTICS and an Editor for the IEEE/RSJ International Conference on Intelligent Robots and Systems and for the IEEE International Conference on Robotics and Automation.