Autonomous Cooperative Mapping of GPS-Denied Cluttered Environments Using Gaussian Process Regression

David Mansfield
School of Engineering
Lancaster University
Lancaster, Lancashire
d.mansfield@lancaster.ac.uk

Nargess Sadeghzadeh-Nokhodberiz
Electrical and Computer Engineering Department
Qom University of Technology
Qom,
sadeghzadeh@qut.ac.ir

Allahyar Montazeri
School of Engineering
Lancaster University
Lancaster, Lancashire
a.montazeri@lancaster.ac.uk

***Abstract*—Multi-agent systems can be used in a range of applications to observe and map spatial-temporal phenomena. In this paper, we have taken the first step to develop a multi-agent environmental monitoring system for fully autonomous exploration and mapping of an unstructured indoor GPS-denied environment. By employing the Gmapping SLAM, the agents cooperatively map a previously unknown environment and explore its entirety. At the same time, the agents are able to successfully map and characterize the temperature distribution inside the room passively using Gaussian Process Regression. The system has been experimentally tested in an indoor cluttered environment, by operation of two Unmanned Ground Vehicles built fully in house. The experimental results show that the proposed system could successfully navigate and explore in the cluttered environment and estimate the spatial distribution of the environment by locating two independent heat sources. It was found that while a passive field prediction approach can approximate the temperature distribution in the room and identify the heat sources, the accuracy of the prediction greatly depends on the proximity of the trajectories that the robots traverse close to the sources.**

***Index Terms*—Cooperative Robots, Multi-agent Systems, GPS Denied, Autonomous Robots, UGV, SLAM, Environmental Monitoring**

I. Introduction

The task of Environmental Monitoring (EM) is becoming more crucial in the modern world. It has a part to play in search and rescue [1], pollution, wildlife and habitat monitoring [2], agriculture [3], nuclear decommissioning [4] and many more. It is often the case that the environments these applications pose are inherently dangerous to humans. Autonomous systems are often cost-effective and low-risk making them attractive solutions. Unmanned Ground Vehicles (UGVs) are ideal choices for many of these tasks, due to their robustness, long-term operation and low power requirements. Furthermore, cooperative multi-agent systems (MAS) offer a robust framework for solving complex problems across various domains in EM. MAS offer redundancy, fault tolerance, parallel processing, reduced operation times and larger operation scales.

It is often the case in EM applications that the agent will be tasked with modeling a spatially distributed environmental phenomena. This could be a gas leak, temperature field, radiation source, pollution concentration or many other things. Furthermore, many of these operation environments encountered in EM are GPS-denied, which means agents need to provide another form of localization based on the environment around them. One common approach is using Simultaneous Localization and Mapping (SLAM). The lack of GPS also makes it harder for agents to cooperate since, unless initial positions are measured before the mission starts, they have no knowledge of their position relative to other agents and hence must use data collected during a mission to calculate a global coordinate frame to facilitate cooperation.

To explore autonomously, agents need to be able to generate waypoints and paths that avoid collisions and are constructive to the mission goal. In EM these paths can either take an active or passive approach to EM. Active approaches are when agents navigate with the goal of maximizing the accuracy of the target phenomenon they are modeling whereas in passive approaches the field sampling is done as the agent explores with respect to another criteria. For example, coverage control is the problem of mapping an environment by ensuring sensor measurements are taken evenly across the whole region of interest. Converge can be performed using fixed search patterns, sometimes called 'lawnmower' algorithms but for a predetermined path to optimally cover the space prior information on the operation environment is needed. Nonetheless, in autonomous SLAM approaches where no priori information about the environment is available, the paths are generated to produce a fully formed map of the physical environment and the waypoints are selected by examining unobserved space. Whether or not one decides to control their agents with a fixed pilot, a fixed search pattern or an autonomous search based on exploration or model optimization depends on the demands and goals of each specific application.

In this paper, we investigated the feasibility of using a multi-agent system of UGVs for a passive EM task in an experimental setting. The system consists of a pair of cooperative UGVs that can autonomously explore and map a unknown and cluttered environment, collaborate on a global map and passively model the temperature distribution. The results show that the trajectories generated for exploration and mapping of the physical environment do not always lead to consistent predictions of the target environmental phenomena but can be used to provide a realistic indication of its distribution.

The remainder of the paper is organized as follows Section II reviews the related work and gives context to the potential applications. Section III formulates the GPR used in this paper and describes how it is trained to model the field. Next, Section IV discusses how SLAM is used to predict the robot's position and orientation and map the physical environment. The process for frontier exploration, path planning and collision avoidance and map merging is discussed. Then, in Section VI the experimental results are presented and discussed. Finally concluding remarks are given in Section VII

II. Related Work

GPS denied navigation is required in numerous environments such as dense urban areas, any indoor environment, underground and underwater. Furthermore, the places where autonomous surveys are most needed are often dangerous for humans, meaning that these environments can be unknown before the mission starts. Perhaps the most ubiquitous approach for solving this problem is Simultaneous Localization And Mapping (SLAM) [5], [6]. But there are other approaches that do not use a map [7], [8]. Multi-agent SLAM systems can be used to reduce mission times but this requires communication and the fusion of each agent's observations. In situations when the initial correspondence is unknown, agents must establish transforms to a global coordinate frame. This is usually done by matching features and landmarks from overlapping parts of local maps [9]. However, this is not a trivial problem as these maps may only partially overlap and the quality of each map can vary from the effect of many factors, including sensors, memory capacity and dynamic environments. In general, there are two kinds of approaches to fusing two maps, direct approaches and feature-based approaches. However, direct approaches are rarely used in practice as they involve examining each pixel in all local map instances which becomes very computationally demanding [10]. In [11], the overlap between RGB-D SLAM maps is detected by using a Bag of Words (BoW) technique that reflects the similarity between two keyframes. All matches above a threshold are then assessed over a number of consecutive keyframes. This helps to avoid false positives in feature mapping. This problem is also addressed for robots in motion in [12] who impose on geometric constrains to improve the accuracy of the feature matching between keyframes.

UGVs are popular for GPS-denied environmental characterization and mapping. In [13] an autonomous UGV was used for radiation mapping, discrete source seeking and inspection of a region of interest. The UGV was fitted with two scintillation detectors one of which was directional. The directional sensor is useful for directing the movement of the robot, especially for source-seeking applications. A popular alternative to UGVs is Unmanned Aerial Vehicles (UAVs), which in the context of indoor applications are very often multi-rotor models [14]. UAVs offer the advantage of 3D movement at the cost of much shorter operation times. In [15] a UAV is presented for navigation in GPS-denied environments, the system is implemented on the Robot Operating System (ROS) and tested in Gazebo. This is a comprehensive simulation platform for testing and development of UAVs in GPS denied environments that also extends to multi-agent simulations [16].

One method to model an environmental phenomena is to combine physical measurements with a mathematical model [17]. But this ultimately restricts the prediction by making assumptions on the structure of the field and, by proxy, parts of the environment. An alternative approach to radiation mapping which does not assume a certain functional form is achieved using regression models. Gaussian Process Regression (GPR) is one such option that is quickly becoming a standard choice for the modelling of spatially distributed data due to its generality, flexibility and accuracy across a wide variety of applications [18]. The authors of [19] applied GP regression for the reconstruction of multiple radiation sources. The Gaussian process (GP) was built offline after the measurements were collected by a robotic arm. GPs can be used with autonomous agents to model all kinds of spatial (and non-spatial) data, for example, it has been applied to terrain mapping and predicting dangerous regions for navigation to improve safety [20].

The authors of [21] use a UGV as a mobile sensor to model two different radiation sources both mapping the physical environment and creating a radiation model using Gaussian process regression. The robots were not autonomous and their trajectory was controlled remotely by a human in a safe location. One perk of using GPs is that they provide a measure of uncertainty with each prediction which can be leveraged for navigational purposes, allowing agents to move in ways that prioritize the cultivation of an accurate predictive model. An example application of using UGVs to optimize GPR for accurate radiation modelling is shown in [22]. The authors employ an autonomous UGV that models the radiation dose rate distribution via GPR and finds the next best measurement position using a differential evolution algorithm driven by the entropy of the GP prediction.

III. Gaussian Process Regression

Suppose robot ri takes a set $\boldsymbol{Y}$ of N noisy observations $\tilde{T}_{ri}$ of the room temperature at corresponding set $\boldsymbol{X}$ of positions $\boldsymbol{q}_{ri}$ predicted by the SLAM algorithm. Such that $\tilde{T}_{ri} = T_{ri} + \sigma_n$. Where T_{ri} is the true temperature reading and σ_n is zero-mean Gaussian noise. The unknown target environmental field $f(x)$ is modelled using Gaussian Process Regression (GPR). GPR is a powerful non-parametric Bayesian approach used for regression analysis. A Gaussian Process (GP) is a collection of random variables, any number of which has a joint Gaussian distribution. We can train a GP to make predictions $f^*(x)$ of the underlying field $f(x)$. GPs provide probabilistic predictions allowing one to compute a measure of uncertainty with each prediction. A GP is entirely defined by a mean function $\mu(\boldsymbol{x})$ and a covariance function $\boldsymbol{C}(\boldsymbol{x}, \boldsymbol{x'})$:

$$f(\boldsymbol{x}) \sim \mathcal{GP}\left(\mu(\boldsymbol{x}), \boldsymbol{C}(\boldsymbol{x}, \boldsymbol{x'})\right). \tag{1}$$

where (x) is a set of input locations. GPR is useful for spatial data as the covariance function $\boldsymbol{C}(\boldsymbol{x}, \boldsymbol{x'})$ is chosen as a kernel function $\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x'})$. Many kernels naturally exhibit spatial correlation and hence work well with spatially distributed samples of environmental phenomena. There are a popular few choices for kernel functions that work well for similar applications. In this work, the chosen kernel was the Matern kernel with $\nu = \frac{3}{2}$ which is a hyperparameter that controls the kernel's smoothness. The Matern kernel is given by [23]:

$$\boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x'}) = \sigma_f^2 \left(1 + \frac{\sqrt{3}r}{\sigma_l}\right) exp\left(-\frac{\sqrt{3}r}{\sigma_l}\right), \tag{2}$$

where r is the Euclidean distance between x and x', σ_f and σ_l are the parameters that describe the correlation between inputs. The tuning of these hyperparameters is done by minimizing the marginal log-likelihood function which describes how well the predicted data fits the observations. The marginal log-likelihood is given by:

$$\begin{aligned} logp(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta}) = &-\frac{1}{2}\boldsymbol{Y}^T(\boldsymbol{K} + \sigma_\epsilon^2\boldsymbol{I})^{-1}\boldsymbol{Y} \\ &-\frac{1}{2}log|\boldsymbol{K} + \sigma_\epsilon^2\boldsymbol{I}| - \frac{N}{2}log(2\pi), \end{aligned} \tag{3}$$

where $\boldsymbol{I}$ is the identity matrix and σ_{ϵ^2} is a zero-mean Gaussian noise. The process of finding the hyperparameters is referred to as training the GP. Once trained we can use the GP to make predictions of target phenomena μ_{f^*} at a set of test locations $\boldsymbol{X}^*$ with uncertainty measurements Σ_{f^*} using the following equations:

$$\mu_{f^*} = \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}^*)\boldsymbol{K}^{-1}(\boldsymbol{X}, \boldsymbol{X})\boldsymbol{Y} \tag{4}$$

and

$$\Sigma_{f^*} = \boldsymbol{K}(\boldsymbol{X}^*, \boldsymbol{X}^*) - \boldsymbol{K}(\boldsymbol{X}^*, \boldsymbol{X})\hat{\boldsymbol{K}}^{-1}(\boldsymbol{X}, \boldsymbol{X})\boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X^*}), \tag{5}$$

where $\hat{\boldsymbol{K}}(\boldsymbol{X}, \boldsymbol{X}) = \boldsymbol{K}(\boldsymbol{X}, \boldsymbol{X}) + \sigma_\epsilon^2\boldsymbol{I}$.

IV. SLAM AND NAVIGATION

The UGVs used in this paper have differential drive kinematics and its position and orientation $\boldsymbol{q}_{ri}(K) = [x_{ri}(k), y_{ri}(k), \theta_{ri}(k)]^T$ can be expressed with the following equations:

$$\begin{bmatrix} x_{ri}(k+1) \\ y_{ri}(k+1) \\ \theta_{ri}(k+1) \end{bmatrix} = \begin{bmatrix} x_{ri}(k) + v_{ri}(k)cos(\theta_{ri}(k))\Delta_t \\ y_{ri}(k) + v_{ri}(k)sin(\theta_{ri}(k))\Delta_t \\ \theta_{ri}(k) + \omega_{ri}(k)\Delta_t \end{bmatrix} + \epsilon(k) \tag{6}$$

where $x_{ri}(k), y_{ri}(k), \theta_{ri}(k)$ are the x-coordinate, y-coordinate and corresponding direction with respect to the x-axis of robot ri at sample time k respectively, and v_{ri} and ω_{ri} are the linear velocity and the angular velocity of robot ri respectively.Δ_t is the sampling interval and $\epsilon(k)$ is zero-mean non-Gaussian process noise with a known probability density function. The SLAM method utilized in this system uses Rao-Blackwellized particle filters (RBPF) to improve grid-based Fast-SLAM [24]–[26]. In particle filter Fast-SLAM the robot's belief of its state is represented as a set of particles, which each represent a possible state with a certain weight that indicates the likelihood of that state being correct. Using odometery $u_{ri}(0:k)$ found from the motion model (Equation 6) and the robots observations $z_{ri}(1:k)$, RBPF can be used to first predict a posterior $p(x_{ri}(1:k)|z_{ri}(1:k), u_{ri}(0:k))$ over potential robot trajectories $x_{ri}(1:k)$. The simplified general steps of the employed method for each robot are summarized in the following:

1) generate path particles using the probabilistic odometry motion model as presented in (6), that is q_{ri}^j $j = 1, \ldots N$ where N refers to the number of particles.
2) compute the importance weight $w^j(k) = w^j(k-1)p(z(k)|m^j(k-1), q_{ri}^j(k)$ where $z(k)$ is the distance measured by the range sensor, m is the grid map and the likelihood is computed using beam model as presented in [27].
3) normalize the weights and do the resampling to obtain highly weighted particles $q_{ri}^j(k)$.
4) update the map $m^j(k)$ using scan matching to compute the best match between $z(k)$ and the reference map $m^j(k-1)$ given $q_{ri}^j(k)$. Toward this, the likelihood function can be employed.

It is worth mentioning that more details of the algorithm with improved proposal distribution and adaptive resampling can be found in [25].

Exploration waypoints are generated for each robot from their individual maps and pose estimations using frontier exploration. A frontier is a boundary between explored and unexplored areas. They are found by finding where known areas meet unknown areas inside the map occupancy grid messages. The centre of the frontier is selected as the next waypoint.

To navigate to their waypoints without colliding with obstacles the robots use both a global and local planner. The global planner determines the best path from the robot location to the current frontier waypoint and considers the entire map. The path is found using Dijkstra's algorithm to minimize the cost. The cost is defined using a cost map that considers static obstacles. The local planner is responsible for executing the global path but considering obstacles along the way using a local cost map that considers both static and dynamic obstacles detected by the robot's sensors. The local planner uses the dynamic window approach (DWA) which selects optimal paths by optimizing an objective function in the form:

$$C_{ri} = \alpha_{ri}C_{\theta,ri} + \beta_{ri}C_{dist,ri} + \gamma_{ri}C_{vel,ri} \tag{7}$$

where $C_{\theta,ri}$ is a function of the difference between robot ri's heading and the required heading to reach its current goal, $C_{dist,ri}$ is a function of the distance of robot ri's distance to its current goal and $C_{vel,ri}$ is a function of the relative velocity of robot ri with respect to its current goal. α_{ri}, β_{ri}

and γ_{ri} are weights that determine the relative contribution of each component. To achieve this planning and navigation the ROS navigation stack is used. This is a standard collection of ROS packages that are used together to allow navigation and collision avoidance via the algorithms discussed.

As the agents are exploring their map and pose data in their local coordinate frames are being shared over a wi-fi connection. Since the robots do not know their starting positions relative to one another the transform between them and the global coordinate frame must be established from the collected data in real time. To achieve this ORB (Oriented FAST and Rotated BRIEF) feature extraction is performed on each map occupancy grid. Then in pairwise fashion, the features for each possible pair of maps are compared. If there is a significant overlap, that is, the number of duplicate features between two maps is above a certain threshold the maps are considered to be overlapping. The transform between each pair of grids is calculated using Random Sample Consensus (RANSAC) which is a robust algorithm used to estimate parameters of a set of data that contains outliers. It is a popular choice for feature matching in computer vision and used for many feature matching applications [28]. Next, a spanning tree graph is constructed in which the nodes represent the component maps and an edge between two maps means they have been matched and the transform between that pair has been established. Then finally that tree is traversed and all maps are transformed to the global coordinate frame which can be chosen to be one of the local coordinate frames. The operation of each agent is summarized in Figure 1.

figures/figure1.png

Fig. 1. A block diagram of the robot system.

V. Methods and Materials

The use of UGVs is beneficial, especially in unreachable and physically dangerous environments often encountered in environmental monitoring. The distributed UGVs used in this study were custom-built at Lancaster University and are shown in Figure 2. They are designed with a modular philosophy and work well as a development platform for many multi-agent robotic applications. They are sturdy and robust but made from inexpensive and replaceable components. They work well for proof of concept tests and full algorithm tests alike although for deployment more application-specific hardware may be required.

figures/figure2.png

Fig. 2. Labled photograph of one the Lancaster University UGV.

These two differential drive robots are capable of cooperatively mapping an unknown location using 2D-Lidar SLAM. They can successfully perform static and dynamic obstacle avoidance and explore autonomously. UGVs such as these can safely operate in environments that pose potential harm to humans. Each UGV is powered by the Robot Operating System (ROS) running on a lightweight and low-power Raspberry Pi 4 model B. The motor drivers are controlled by an Arduino UNO and the odometry is provided by the digital motor encoders of each. The system's power supply is a 4-cell 16.8V 6600mah LiPo battery. The agents share map data over the ROS network via a Wi-Fi connection. The data was recorded using rosbag.

figures/figure3.png

Fig. 3. The operation environment showing initial robot positions and heat sources.

Without a priori knowing the relative position of the other agent using scan matching techniques the agents are able to merge their individual SLAM maps and localize their selves w.r.t to one another. This does require some overlap in the sensed areas but makes the system more widely applicable as it is possible that the relative position would be unknown in practical scenarios. The robots were tasked with exploring an unknown environment and measuring temperature as they moved. Temperature, in this case, represents a general environmental field of interest and the same algorithms used here could be applied to radiation, humidity, gas etc with very little or no algorithmic changes. The temperature measurements and corresponding x and y coordinates were passed to the GP regression ROS node that makes a prediction over the given area. To perform GP regression GPYTorch was used. The experiment was carried out in a small seminar room with the furniture arranged to create a cluttered environment for the agent to navigate. The environment was configured in this way to separate the agents initially and to allow full demonstration of the planning and obstacle avoidance capabilities. The room is shown in Figure 3.

VI. Results and Discussion

The robots were tasked with exploring a cluttered environment and mapping the temperature passively. This kind of operation is useful for preliminary monitoring applications like measuring the radiation intensity throughout a nuclear power plant undergoing decommissioning. A challenging and densely cluttered course was made with two heat sources placed inside. It should be noted that due to limited resources, the heat sources used are cheap electric fan heaters and only provide significant temperature changes in very close proximity. That being said under these constrained conditions the GP predictions were able to identify both sources which demonstrates the effectiveness of the system. The environment, initial starting positions and locations of the heat sources are shown in Figure 3. The ambient temperature of the room was 22.4°C.

The robots were successfully able to navigate the room safely without any collisions and successfully planned efficient exploration paths around the room. Exploration and map merging was always completed in under 2 minutes. The results figures/figure4.png

Fig. 4. The GP temperature field predictions and robot trajectories superimposed onto the local robot maps. Heat sources are labelled ”A” and ”B”.

figures/figure5.png

Fig. 5. The local component maps and the merged map at the time of merging.

of the map merging are shown in Figure 5. The maps are very slightly offset which is due to observation noise, differences in each robot’s local map and uncertainties in the global coordinate frame calculation, That being said, the global map is still an accurate representation of the room and would allow a robot to navigate and perform reliable path planning within it which is the main aim. Due to the size of the room each robot had explored most of the environment before map merging. This means that the improvement of efficiency from multi-agent cooperation was very marginal, as each robot only had a small area left to explore. However, the time to merge maps is more closely related to the amount of overlapping features in each component map rather than the extent to which the total environment has been explored. This means that in a larger environment, this system has a much higher potential of increasing efficiency. This insight indicates that for a multi-agent system with unknown initial relative starting locations, the efficiency gain is related to how quickly the two agents meet.

Alongside autonomous exploration, each robot was also tasked with mapping the temperature in the room. There are numerous considerations that need to be taken into account when analysing these results. First and foremost, the temperature sources are inexpensive electric heaters and only provide high-temperature fluctuations in very close proximity and lead to very slight changes in temperature in the surrounding area. This, combined with the fact that the agents were sensing and predicting temperature passively, that is, not adapting their behaviour based on the temperature readings it means that if the agents did not pass the source closely the lack of high-intensity temperature readings will be reflected in the prediction. However, this set-up was chosen as it accurately reflects the sparse nature of many different real environmental scenarios and the temperature sources were still detected despite a low variation. The GP temperature field predictions are shown in Figure 4. These results show the predictions for each agent on its local map as the current system does not collaborate on temperature prediction. One agent successfully identified both heat sources and predicted heat source A to be more intense. On the other hand, the other agent detected a similar heat plume for heat source B but missed heat source A completely. This kind of result was a frequent occurrence. The differences in predictions stem from numerous sources. Firstly passing more closely to a heat source will allow for higher temperature readings and a better chance for the source to be reflected in the prediction. It is also the case that if an agent passes much more closely to one source than the other those high values can saturate the other hot spots if there is a low density of samples with high values near the other source. Finally, the speed and orientation of the agents as they pass through a source will affect the readings as the sensor was located only at the front of the robot.

VII. Conclusion

These results demonstrate the power of GPR for predicting environmental fields under serious constraints. Firstly the target field is very uniform apart from small gradients in very localized areas, the agents were using cheap and inexpensive sensors and finally, the movement of the robots was not performed in such a way as to optimize field predictions. Despite this, the results still give a sensible estimation of the sources. It can also be appreciated that cooperation on field prediction may yield more accurate results again demonstrating the advantages of multi-agent systems. Implementing navigation for field optimization and cooperative temperature field predictions is the subject of future work.

References

[1] S. Waharte and N. Trigoni. Supporting search and rescue operations with uavs. In *2010 International Conference on Emerging Security Technologies*, pages 142–147, 2010.

[2] M. Egerstedt, J.N. Pauli, G. Notomista, and S. Hutchinson. Robot ecology: Constraint-based control design for long duration autonomy. *Annual Reviews in Control*, 46:1–7, 2018.

[3] T. Duckett, S. Pearson, S. Blackmore, B. Grieve, W.-H. Chen, G. Cielniak, J. Cleaversmith, J. Dai, S. Davis, C. Fox, P. From, I. Georgilas, R. Gill, I. Gould, M. Hanheide, A. Hunter, F. Iida, L. Mihalyova, S. Nefti-Meziani, G. Neumann, P. Paoletti, T. Pridmore, D. Ross, M. Smith, M. Stoelen, M. Swainson, S. Wane, P. Wilson, I. Wright, and G.-Z. Yang. Agricultural robotics: The future of robotic agriculture. 2018.

[4] P.G. Martin, S. Kwong, N.T. Smith, Y. Yamashiki, O.D. Payton, F.S. Russell-Pavier, J.S. Fardoulis, D.A. Richards, and T.B. Scott. 3d unmanned aerial vehicle radiation mapping for assessing contaminant distribution and mobility. *International Journal of Applied Earth Observation and Geoinformation*, 52:12–19, 2016.

[5] N Sadeghzadeh-Nokhodberiz, A Can, R Stolkin, and A Montazeri. Dynamics-based modified fast simultaneous localization and mapping for unmanned aerial vehicles with joint inertial sensor bias and drift estimation. *IEEE Access*, 9:120247–120260, 2021.

[6] J.A. Placed, J. Strader, H. Carrillo, N. Atanasov, V. Indelman, L. Carlone, and J.A. Castellanos. A survey on active simultaneous localization and mapping: State of the art and new frontiers, 2023.

[7] D. Mansfield and A. Montazeri. A survey on autonomous environmental monitoring approaches: Towards unifying active sensing and reinforcement learning [preprint]. *Frontiers in Robotics and AI*, 11, 2024.

[8] Y. Chang, Y. Cheng, U. Manzoor, and J. Murray. A review of uav autonomous navigation in gps-denied environments. *Robotics and Autonomous Systems*, 170:104533, 2023.

[9] H. Malakouti-Khah, N. Sadeghzadeh-Nokhodberiz, and A. Montazeri. Simultaneous localization and mapping in a multi-robot system in a dynamic environment with unknown initial correspondence. *Frontiers in Robotics and AI*, 10, 2024.

[10] J. Vaščák and D. Herich. Map merging for multi-robotic applications. In *2023 IEEE 21st World Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pages 000021–000026, 2023.

[11] Tianyun Ma, Tao Zhang, and Shaopeng Li. Multi-robot collaborative slam and scene reconstruction based on rgb-d camera. In *2020 Chinese Automation Congress (CAC)*, pages 139–144, 2020.

[12] Y. Chen, Y. Wang, J. Lin, Z. Chen, and Y. Wang. Multi-robot point cloud map fusion algorithm based on visual slam. In *2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE)*, pages 329–333, 2021.

[13] T. Lazna, O. Fisera, J. Kares, and L. Zalud. Localization of Ionizing Radiation Sources Via an Autonomous Robotic System. *Radiation Protection Dosimetry*, 186(2-3):249–256, 12 2019.

[14] A. Can, I.H. Imran, J. Price, and A. Montazeri. Robust formation control and trajectory tracking of multiple quadrotors using a discrete-time sliding mode control technique. *IFAC-PapersOnLine*, 55(10):2974–2979, 2022. 10th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2022.

[15] A. Can, J. Price, and A. Montazeri. A nonlinear discrete-time sliding mode controller for autonomous navigation of an aerial vehicle using hector slam. *IFAC-PapersOnLine*, 55:2653–2658, 10 2022.

[16] X. Zhu, F. Vanegas, F. Gonzalez, and C. Sanderson. A multi-uav system for exploration and target finding in cluttered and gps-denied environments. In *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 721–729, 2021.

[17] F. Mascarich, M. Kulkarni, P. De Petris, T. Wilson, and K. Alexis. Autonomous mapping and spectroscopic analysis of distributed radiation fields using aerial robots. *Autonomous Robots*, 47:1–22, 11 2022.

[18] A. Viseras and C. Olariu. A general algorithm for exploration with gaussian processes in complex, unknown environments. volume 2015, 05 2015.

[19] D. De Schepper, M. Simons, W. Schroeyers, K. Kellens, and E. Demeester. Learning multiple radiation source distribution models using gaussian processes. In *ISR Europe 2023; 56th International Symposium on Robotics*, pages 432–439, 2023.

[20] M. Budd, B. Lacerda, P. Duckworth, A. West, B. Lennox, and N. Hawes. Markov decision processes with unknown state feature values for safe exploration using gaussian processes. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7344–7350, 2020.

[21] A. West, I. Tsitsimpelis, M. Licata, A. Jazbec, L. Snoj, M. Joyce, and B. Lennox. Use of gaussian process regression for radiation mapping of a nuclear reactor with a mobile robot. *Scientific Reports*, 11, 07 2021.

[22] P.R. Silveira, D. de F. Naiff, C.M.N.A. Pereira, and R. Schirru. Reconstruction of radiation dose rate profiles by autonomous robot with active learning and gaussian process regression. *Annals of Nuclear Energy*, 112:876–886, 2018.

[23] C. E. Rasmussen and C. K. I. Williams. Covariance functions. In *Gaussian Processes for Machine Learning*. The MIT Press, 11 2005.

[24] G. Grisetti, C. Stachniss, and W. Burgard. Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 2432–2437, 2005.

[25] G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, 23(1):34–46, 2007.

[26] M. Montemerlo, S. Thrun, D. Roller, and B. Wegbreit. Fastslam 2.0: an improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, IJCAI'03, page 1151–1156, San Francisco, CA, USA, 2003. Morgan Kaufmann Publishers Inc.

[27] Juan-Antonio Fernández-Madrigal. *Simultaneous Localization and Mapping for Mobile Robots: Introduction and Methods: Introduction and Methods*. IGI global, 2012.

[28] N. Sadeghzadeh-Nokhodberiz, M. Iranshahi, and A. Montazeri. Vision-based particle filtering for quad-copter attitude estimation using multirate delayed measurements. *Frontiers in Robotics and AI*, 10, 2023.