UNIVERSITÄT BONN

Dissertation
zur Erlangung des Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)
Agrar-, Ernährungs- und Ingenieurwissenschaftliche Fakultät
der Rheinischen Friedrich-Wilhelms-Universität Bonn
Institut für Geodäsie und Geoinformation

Integrated Robotic Learning and Planning for UAV-Based Information Gathering in Unknown Environments

von

Julius Rückin

aus

Osnabrück, Germany

Bonn 2025

Referent:
Dr. Marija Popović, Delft University of Technology, Netherlands

1. Korreferent:
Prof. Dr. Cyrill Stachniss, University of Bonn, Germany

2. Korreferent:
Prof. Dr. Ribana Roscher, University of Bonn, Germany

Tag der mündlichen Prüfung: 27. Mai 2025

Angefertigt mit Genehmigung der Agrar-, Ernährungs- und Ingenieurwissenschaftlichen Fakultät der Universität Bonn

Zusammenfassung

Roboter automatisieren zunehmend Aufgaben, die kostenintesiv oder gefährlich sind oder präzise Messungen in großem Maßstab erfordern. Durch die Kombination von Fortschritten in der Hardwareentwicklung und im maschinellen Lernen können Roboter mittels Sensoren Daten sammeln und Messungen automatisiert interpretieren, um Informationen, z. B. zur Analyse von Nutzpflanzen für Züchter oder von Städten zu Zwecken des Katastrophenmanagements, zu erfassen. Klassische Missionen zur Informationserfassung setzen oft eine bekannte Umgebung voraus und sammeln traditionellerweise Daten entlang vorprogrammierter Pfade. Für die Datenerfassung in unbekannten Umgebungen ist häufig eine manuelle Überwachung oder Bedienung erforderlich, was den Grad der Automatisierung einschränkt. Um das Potenzial, Informationen zu erfassen voll auszuschöpfen, benötigen wir Algorithmen, die es dem Roboter ermöglichen, Aktionen direkt an Bord zu planen und auszuführen. Um ihr Verhalten flexibel anzupassen, müssen Roboter die unbekannte Umgebung autonom erkunden und dabei Ressourcenbeschränkungen wie z.B. begrenzte Energie- und Rechenkapazitäten eines unbemannten Luftfahrzeugs (UAV) berücksichtigen. Eine wichtige Fähigkeit besteht darin, unter Unsicherheit Entscheidungen darüber zu treffen, wo neue Daten über interessante Regionen gesammelt werden sollen, z. B. über Krankheitsherde in einem Erntefeld, basierend auf dem unvollständigen Umgebungsverständnis während einer Mission. Diese Aufgabe wird auch als adaptives Wegplanungsproblem bezeichnet.

Die Hauptbeiträge dieser Arbeit sind neuartige, lernbasierte, adaptive Wegplanungsmethoden für die UAV-basierte Informationserfassung in unbekannten Umgebungen. Unsere Methoden leiten ein ressourcenbeschränktes UAV in Regionen, in denen es informative neue Messungen sammeln könnte, um sein aktuelles Umgebungsverständnis zu verbessern und so effizient Informationen zu sammeln.

Zuerst stellen wir eine neue adaptive Wegplanungsmethode vor, die klassische Wegplanungsansätze mit Fortschritten im Bereich des bestärkenden Lernens kombiniert, um Strategien zur Informationserfassung zu trainieren. Wir entwickeln ein auf einer Baumsuche basierendes Wegplanungsverfahren, das durch gelern-

te neuronale Netze gesteuert wird. Unsere lernbasierte Methode beschleunigt die Wegplanung während der Mission auf einem ressourcenbeschränkten UAV im Vergleich zu nicht lernbasierten, rechenintensiven Planungsmethoden.

Die meisten Wegplanungsmethoden werden für bestimmte Kartenrepräsentationen der Umgebung entwickelt und trainiert. Dadurch sind sie entweder für die Erfassung von Informationen mit kontinuierlichen Werten, wie z. B. der Oberflächentemperatur, oder diskreten Werten, wie z. B. der Segmentierung von Unkraut und Pflanzen, geeignet. Unsere zweite Methode löst diese Einschränkung durch eine neue Formulierung des adaptiven Wegplanungsproblems auf, die Missionen mit beliebigen zu beobachtenden Informationen vereinheitlicht. Basierend auf unserer Formulierung trainieren wir eine karten-agnostische Strategie zur Informationserfassung in Umweltbeobachtungsmissionen mittels bestärkenden Lernens. Zusätzlich vereinheitlicht unsere Formulierung vorherige Wegplanungsmethoden.

In Missionen, bei denen Bilder mithilfe von mehrschichtigen Bildverarbeitungsmodellen semantisch interpretiert werden, verschlechtern sich Modellvorhersagen in unbekannten Umgebungen oft. Daher sind kostenintensive manuelle Annotation der gesammelten Bilder erforderlich, um das Bildverarbeitungsmodell zu verbessern. Um das semantische Sehen eines UAV in unbekannten Umgebungen zu verbessern, stellen wir ein neues adaptives Wegplanungssystem für das aktive Lernen von semantischen Segmentierungsmodellen vor. Im Vergleich zu aktuellen nicht-adaptiven Kampagnen zur Sammlung von Trainingsdaten verbessert unser System die semantische Segmentierung schneller und reduziert gleichzeitig die Anzahl der benötigten manuell annotierten Bilder.

Unsere vierte Methode ist ein neues halbüberwachtes Lernverfahren zur Verbesserung des semantischen Sehens in unbekannten Umgebungen, um den Aufwand der manuellen Annotation weiter zu verringern. Für das Modelltraining kombinieren wir eine kleine Menge an manuell annotierten Bildpixeln mit automatisch annotierten Pixeln, die auf der semantischen Umgebungskarte basieren, die das UAV online erstellt. Unsere Methode benötigt weniger als ein Prozent der manuell annotierten Pixel, um eine ähnlich akkurate semantische Segmentierung zu erzielen wie die auf der manuellen Annotation aller Bildpixel basierende Methode.

Die in dieser Dissertation vorgestellten Methoden wurden in begutachteten Konferenzbeiträgen und Zeitschriftenartikeln veröffentlicht. Unser Beitrag, in dem wir offenen Forschungsfragen des aktiven Lernens mittels adaptiver Wegplanung diskutieren, erhielt den Preis für den besten Beitrag in dem IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems Workshop on Label Efficient Learning Paradigms for Autonomy at Scale. Des Weiteren haben wir Implementierungen aller Methoden der Forschungsgemeinschaft als Open Source zur Verfügung gestellt.

Abstract

Robots increasingly automate tasks that are costly, dangerous, or require precise measurements at scale. Combining advances in hardware with recent progress in machine learning-based computer vision enables robots to collect data with onboard sensors and interpret measurements to gather information, e.g. monitoring crop conditions for breeders or cities for disaster management. Classical information-gathering missions might require the environment to be known before deployment and traditionally execute pre-programmed paths for robotic data collection. In unknown environments, robot autonomy is often limited by the need for human supervision or operation. To fully leverage the information-gathering potential, we need algorithms that enable the robot to plan actions onboard during its deployment. Mainly, robots must autonomously collect information and adapt their behaviour online in the unknown environment while considering onboard resource constraints, such as the limited energy and compute power of unmanned aerial vehicles (UAVs). A key aspect is making decisions under uncertainty where to collect new informative data about areas of interest, e.g. disease hotspots in a crop field, based on the robot's incomplete understanding of the environment during a mission. This task is also known as the adaptive informative path planning problem.

The main contributions of this thesis are novel learning-based adaptive informative path planning approaches for UAV-based information gathering in unknown environments. Our approaches guide a resource-constrained UAV towards areas where it could collect informative new measurements to enhance its current understanding of the environment in which it operates, thus increasing its efficiency in information gathering within the mission constraints.

The first approach is a new adaptive informative path planning method combining classical robotic path planning with reinforcement learning to train strategies for information-gathering missions. We connect the adaptive informative path planning problem and the general reinforcement learning problem. Using this connection, we develop a tree search-based replanning procedure guided by learned neural networks. Our learning-based method accelerates path replanning during deployment on a resource-constrained UAV compared to previous non-learning-based adaptive informative path planning methods.

Most adaptive informative path planning methods are designed and trained for certain environmental map representations. This makes them applicable to either monitor continuous-valued information, such as surface temperature, or discrete-valued information, such as segmentation of weeds and crops. However, these methods cannot be applied to changing to-be-monitored information without re-designing or re-training planning strategies. The second approach addresses this limitation by introducing a novel mathematical formulation of the adaptive informative path planning problem that unifies missions with arbitrary to-be-monitored environment information. Using our formulation, we train a map-agnostic information-gathering strategy with reinforcement learning for environmental monitoring missions. Moreover, our formulation unifies previously developed adaptive informative path planning methods.

In missions that require semantic interpretation of images using deep learning-based vision models, the model's prediction performance and, hence, the UAV's efficiency in information gathering typically degrade in unknown environments. Thus, these missions require costly human annotations of collected images to re-train and improve vision models. We propose a novel adaptive informative path planning framework for active learning of semantic segmentation models to improve a UAV's semantic vision in unknown environments while reducing human annotations. Our key insight is to link model uncertainty measures from active learning to the information-gathering planning objective. Our framework improves the semantic segmentation performance faster while drastically reducing the number of human-labelled images required to train the semantic vision model compared to non-adaptive exhaustive training data collection campaigns.

Lastly, our fourth approach is a novel semi-supervised learning method for improving semantic vision in unknown environments to further reduce human labelling efforts. For model training, we combine a sparse set of image pixels selected for human labelling with automatically labelled pixels based on the semantic environment map the UAV builds during deployment. Our new pixel selection method for human labelling outperforms state-of-the-art methods. Our automatically generated labels further improve model performance. Overall, our method requires less than one per cent of the human-labelled pixels to maintain semantic segmentation performance similar to exhaustively labelling all pixels.

Our approaches proposed in this thesis have been published in peer-reviewed conferences and journals. Our paper discussing open research questions in active learning of robot vision received the best paper award at the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems as part of the Workshop on Label Efficient Learning Paradigms for Autonomy at Scale. We made all method implementations available as open source to foster further research.

Acknowledgements

First, I would like to thank Dr. Marija Popović for giving me the opportunity to pursue my doctoral studies in her group at the University of Bonn, for giving me the freedom to find and do research on topics that interested me most, and for always providing valuable feedback on my work. I would also like to thank Prof. Dr. Cyrill Stachniss for the many collaborations during my doctoral studies, for allowing me to join his group during my final year, and for his supportive guidance. Thank you to Thomas Läbe and Sonja de Vries for their support with technical and administrative questions and all my lab colleagues for making it a nice and easy start in your group.

Furthermore, I would like to thank my collaborators. Thank you to Liren Jin and Federico Magistri for being among my closest collaborators. Thank you to Prof. Dr. Eduardo Montijano and David Morilla-Cabello for our fruitful collaboration on map-agnostic adaptive informative path planning. Thank you to Prof. Dr. Mykel J. Kochenderfer and Dr. Joshua Ott for our insightful collaboration on the learning-based adaptive informative path planning survey. A special thanks to Gianmarco Roggiolani and PD Dr. Jens Behley for including me in their research projects and for being wonderful lab colleagues. Thank you to Jonas Westheider, Tobias Zaenker, Rohit Menon, Prof. Dr. Maren Bennewitz, Dr. Stefan H. Kiss, Prof. Dr. Teresa Vidal-Calleja, Alireza Ahmadi, Dr. Michael Halstead, and Prof. Dr. Chris McCool for various collaborations. Additionally, I would like to thank Franziska Külbel for her linguistic review of this thesis.

Finally, I cannot thank my longtime friends from my Bachelor's and Master's studies enough, people who became friends during my doctoral studies in Bonn, and my family for the support I received. In particular, I would like to thank my sister and Franziska Külbel for their support during the last few years.

The work presented in this thesis has been funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy, EXC-2070 – 390732324 (PhenoRob). The financial support of the DFG through the PhenoRob project is gratefully acknowledged.

Contents

Chapter 1

Introduction

TREMENDOUS advances in the fields of robotics and machine learning have sparked a new wave of robotic systems that aim to automate tasks executed in environments other than carefully controlled lab conditions. These robots are deployed in various environments to execute costly, dangerous tasks, as well as those which require precise measurements at scale. Robots are used to monitor crop conditions [95, 172, 179] to deliver valuable information for crop breeders, manage weeds in arable fields to reduce agrochemical usage [1, 104], and are used in urban planning to analyse land use [75, 133, 166] or in industrial sites to inspect infrastructure [86, 186]. In addition, remarkable progress in deep learning-based vision techniques [59, 81, 168] enables robots to automatically interpret data collected with onboard sensors, e.g. segmenting imagery of arable fields into different semantics, such as weed and crop [109, 140, 176], to extract task-relevant information from collected data.

Although such vision methods allow automated interpretation of collected sensor data, many of today's robotic systems act mainly as passive data collection devices. Classical robotic systems require human supervision or operation to collect data in an unknown environment. If the environment is known before deployment, traditional robotic systems often execute pre-programmed paths along which the data is collected [48]. To fully leverage the potential of information gathering in unknown environments, a robot needs to be equipped with the ability to actively plan its next actions directly onboard during deployment. In this way, the robot can flexibly adapt its behaviour online based on its current understanding of the environment, which allows for gathering more information in less time during a mission [64]. To this end, the robot needs to autonomously explore initially unknown environments while considering the platform's resource constraints, such as limited energy and compute power of unmanned aerial vehicles (UAVs). To achieve this, a key algorithmic capability is to decide where to move next to efficiently collect informative data about task-relevant areas of

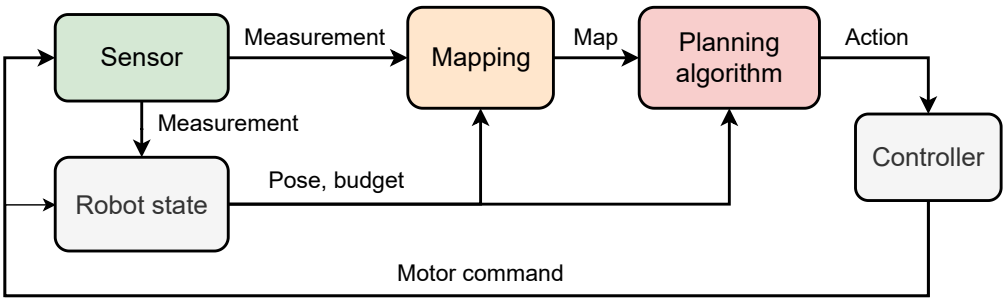


Figure 1.1: Overview of a robotic system for an onboard perception-action loop. The robot is equipped with sensors to collect measurements, which are fused into an environment map. Based on the current map and the robot's state, the planning algorithm decides which action to execute next. The controller computes the low-level actuation commands to execute the action.

interest based on the robot's current noisy and incomplete understanding of the environment during deployment [106, 127], such as finding areas of high weed pressure in arable fields to inform in-field management.

Robotic information gathering is closely connected to active perception, requiring a percept to reason about how to change a passive sensor's state to gather goal-specific information, i.e. to perform active sensing [5]. The robotic system relies on a perception-action loop to adaptively plan the next actions for active sensing. A typical system incorporating this perception-action loop is depicted in Fig. 1.1. The robot is equipped with an onboard sensor to collect measurements, which are fused into an environment map based on the robot's current pose and previously collected measurements. The planning algorithm uses the environment map and the current robot state to reason about and decide on which action, e.g. movement, to execute next. The controller is responsible for the low-level actuation of motors to execute the planned action. Classical planning algorithms enable the robot to plan its path towards a goal position [58, 73, 87]. Although these planning algorithms tell the robot how to navigate towards a position, they cannot plan where to go next on a higher level of reasoning, rendering them impractical for fully autonomous guidance in unknown environments.

The focus of this thesis is to develop novel planning algorithms that efficiently guide a resource-constrained robot online in an initially unknown environment towards areas where the robot could potentially collect informative new sensor measurements to improve its understanding of the environment. In the literature, this is known as the adaptive informative path planning (IPP) problem [64, 127]. Non-adaptive IPP methods pre-plan informative paths prior to a mission and execute these fixed paths during deployment [106]. These methods perform well in exploring unknown environments with limited resources but cannot adapt their behaviour during a mission based on the robot's evolving understanding of the environment. Thus, non-adaptive IPP methods may be limited in terms of

information-gathering efficiency in scenarios where measurement uncertainty depends on the observed data or in which information is non-uniformly distributed, e.g. if a user is interested in certain non-uniformly distributed environment features, such as hotspots of weed pressure in an arable field. In contrast, adaptive IPP methods replan informative paths onboard during deployment based on the robot's evolving understanding of the environment. This thesis focuses on new adaptive IPP methods in the context of information-gathering missions over 2D flat terrains deployed on UAVs. An example of such a mission is a UAV with limited compute resources and flight time, monitoring unknown arable fields and tasked to precisely map surface temperature hotspots as these areas might indicate plant drought stress. Our proposed methods could be adopted to robot platforms and mission goals different from the ones showcased in this thesis.

Despite an increasing research effort in the development of adaptive IPP algorithms and their encouraging higher information-gathering efficiency than non-adaptive pre-programmed paths, classical adaptive IPP methods still show limitations. First, most adaptive IPP methods solve the planning problems described above in a computationally expensive fashion. This leads to high onboard compute requirements, slow replanning of future paths, or a sacrifice of path planning performance. Thus, the adaptivity and practicality of classical adaptive IPP algorithms for onboard deployment on resource-constrained mobile robots are limited. Second, most adaptive IPP methods assume a reliable performance of onboard sensors used to interpret and extract information from incoming measurements. However, with today's widespread adoption of deep learning-based computer vision models, e.g. to semantically segment buildings, streets, and vegetation in collected images for urban planning purposes, this assumption is often not valid. As the environment for robot deployment is initially unknown, the incoming sensor measurements often deviate from those the deep learning-based vision system was trained on. In these cases, the prediction quality of these kinds of sensors often drastically degrades. Thus, the amount of information extracted from individual sensor measurements is limited by the vision model's performance, leading to an overall degraded robotic information-gathering efficiency.

This thesis will tackle these outlined limitations and propose new, more efficient approaches to these problems. Overall, this thesis answers the following three research questions in the context of adaptive IPP for terrain monitoring in unknown environments as depicted in Fig. 1.2:

1. How to increase the compute efficiency of adaptive IPP algorithms without sacrificing planning performance in information-gathering missions?

2. How to improve deep learning-based vision models in unknown environments while minimising the amount of new human-labelled training images?

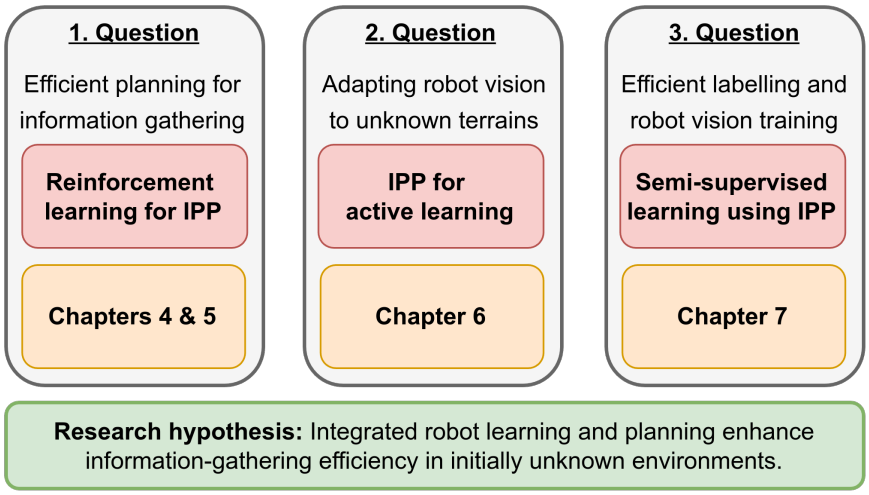


Figure 1.2: Overview of research questions (grey), hypothesis (green), main contributions (red), and structure (orange) of this thesis. In Chap. 4 and Chap. 5, we propose new adaptive IPP methods using reinforcement learning. In Chap. 6, we introduce an adaptive IPP framework for active learning of deep learning-based robot vision. In Chap. 7, we develop a new semi-supervised learning method for robot vision. Our research hypothesis is that combining robot learning and planning improves the efficiency of robotic information gathering.

3. Which image queries should be labelled by a human annotator, and which vision learning signals can be derived from the robot's actively improving understanding of the environment to make labelling more efficient?

We propose novel approaches for adaptive IPP combining advances in machine learning with classical robotic planning approaches. *Our main research hypothesis is that these integrated robot learning and planning approaches can enhance information-gathering efficiency in UAV-based terrain monitoring missions.* In Chap. 4 and Chap. 5, we propose new learning-based adaptive IPP methods using reinforcement learning to tackle the first question. In Chap. 6, we introduce a novel adaptive IPP framework for active learning of deep learning-based vision sensors to answer the second question. In response to the third question, in Chap. 7, we develop a new semi-supervised learning method for robotic semantic vision using adaptive IPP to drastically reduce human labelling efforts.

1.1 Main Contributions

The main contributions of this thesis are novel approaches for adaptive IPP in UAV-based terrain monitoring missions combining robot learning and planning. In Chap. 2, we review existing IPP approaches and discuss the advantages of our proposed methods. Our methods presented in Chap. 4 and Chap. 5 leverage advances in reinforcement learning (RL). To aid understanding of these adaptive IPP methods, in Chap. 3, we introduce basic RL terminology and algorithms.

In Chap. 4, we introduce a novel adaptive IPP algorithm for information gathering in UAV-based terrain monitoring missions. The robot is tasked to map continuous-valued features over 2D flat terrains, such as signal strength or surface temperature. It aims to quickly find and precisely map user-defined areas of interest, e.g. areas of high surface temperature in arable fields, performing online replanning of paths on a compute- and flight time-constrained UAV. Specifically, we combine recent advances in RL for playing board games with classical sampling-based robotic path planning algorithms into a novel RL-based adaptive IPP approach. Our approach shows high information-gathering efficiency in terms of map uncertainty reduction over time while drastically reducing replanning run time compared to previous non-learning-based adaptive IPP methods on compute-constrained platforms. These results could facilitate deploying adaptive IPP methods on resource-constrained mobile robots.

Despite progress in learning-based adaptive IPP methods, these approaches are often not applicable beyond the specific mission characteristics they were trained on. First, learning-based methods assume to-be-mapped terrain features either to be continuous-valued, e.g. surface temperature, or discrete-valued, e.g. weed-crop classification. Mapping continuous- or discrete-valued terrain features requires different map representations that are directly input to the learned planning strategies. Thus, these strategies are only applicable to the map representation on which they were trained. Second, planning strategies are trained on static user-defined parameters, such as sets of feature values qualifying a terrain area to be of interest for precise mapping, e.g. high surface temperature values above a certain threshold. Thus, these approaches tend to overfit a single user-defined criterion for areas of interest, dropping in performance as user interests change. Overall, these assumptions require laborious re-design and time-consuming re-training of learning-based methods as mission characteristics change.

In Chap. 5, we address these limitations and propose a novel map-agnostic adaptive IPP formulation for terrain monitoring. We derive a new planning state representation as input to the adaptive IPP strategy that unifies varying map representations. Based on this map-agnostic state and a new reward function, we train a single adaptive IPP strategy using RL that is applicable to continuous- and discrete-valued terrain feature monitoring missions with different user-defined areas of interest. Our planning strategy maintains information gathering performance on various terrain monitoring missions compared to state-of-the-art map-specifically designed or trained adaptive IPP methods. Hence, our approach facilitates deploying learned adaptive IPP strategies without re-designing and re-training for specific mission characteristics. Furthermore, our map-agnostic formulation integrates with and unifies state-of-the-art non-learning-based adaptive IPP methods while maintaining their performance.

Chap. 6 answers the question of how to adapt deep learning-based semantic vision models to unknown environments while minimising the number of costly human-labelled images. We introduce a novel adaptive IPP framework for active learning in semantic mapping missions, e.g. monitoring of buildings, streets and vegetation for urban planning purposes. Active learning is a branch of computer vision research that aims to develop algorithms selecting a minimal number of to-be-labelled data points from an existing large pool of unlabelled data maximally improving the vision model's performance. In contrast, we directly deploy the robot in the initially unknown environment without prior access to such a data pool. We develop adaptive IPP algorithms to target areas of informative new training data that potentially maximise semantic segmentation performance upon vision model retraining. We attain this targeted planning behaviour by proposing novel robotic planning objectives that combine ideas from active learning with adaptive IPP. Our framework maximises the deep learning-based vision system's performance while minimising the number of human-labelled images compared to state-of-the-art planning methods for active learning across various environments with vastly different semantics. Thus, our approach minimises the human labelling effort required to adapt the robot's vision in unseen environments.

Finally, in Chap. 7, we extend our adaptive IPP framework for active learning by developing a new semi-supervised learning approach to reduce human labelling efforts further. Instead of labelling all pixels of an image manually, we propose a new selection criterion to choose a sparse set of pixels from collected images to be labelled by a human annotator. Additionally, we automatically create consistent semantic labels from the robot's online-built semantic environment map to further improve the model's performance without additional human labels. Our semi-supervised approach to adaptive IPP for active learning drastically reduces the human labelling effort of state-of-the-art fully supervised approaches while maintaining the vision model's performance. Furthermore, it outperforms self-supervised approaches in semantic terrain mapping missions. Hence, our approach answers the third research question of how to leverage the robot's understanding of the environment and which labelling queries to ask a human annotator to most efficiently adapt the robot's vision to unseen terrains.

In sum, this thesis presents novel approaches for adaptive IPP in the context of UAV-based terrain monitoring missions. We make several key contributions to the field of robotic information gathering and to the field of active learning for robot vision. Our approaches show how to combine adaptive IPP with RL and how to combine adaptive IPP methods with semi-supervised active learning to improve the information-gathering efficiency of prior state-of-the-art methods. We made implementations of all presented methods publicly available to foster further research. The links to the implementations are listed in Sec. 1.2.

1.2 Publications

Parts of this thesis have been published in the following peer-reviewed conference and journal articles, to which I have been the main contributor:

- Julius Rückin, David Morilla-Cabello, Cyrill Stachniss, Eduardo Montijano, and Marija Popović. Towards Map-Agnostic Policies for Adaptive Informative Path Planning. *IEEE Robotics and Automation Letters (RA-L)*, 10(5):5114–5121, 2025. doi: 10.1109/LRA.2025.3557233.

- Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Active Learning of Robot Vision Using Adaptive Path Planning. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS) Workshop on Label Efficient Learning Paradigms for Autonomy at Scale*, 2024

- Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Semi-Supervised Active Learning for Semantic Segmentation in Unknown Environments Using Informative Path Planning. *IEEE Robotics and Automation Letters (RA-L)*, 9(3):2662–2669, 2024.
  doi: 10.1109/LRA.2024.3359970.

- Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. An Informative Path Planning Framework for Active Learning in UAV-Based Semantic Mapping. *IEEE Trans. on Robotics (TRO)*, 39(6):4279–4296, 2023. doi: 10.1109/TRO.2023.3313811.

- Julius Rückin, Liren Jin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Informative Path Planning for Active Learning in Aerial Semantic Mapping. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022. doi: 10.1109/IROS47612.2022.9981738.

- Julius Rückin, Liren Jin, and Marija Popović. Adaptive Informative Path Planning Using Deep Reinforcement Learning for UAV-based Active Sensing. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022. doi: 10.1109/ICRA46639.2022.9812025.

As part of my doctoral studies, I have contributed to the following peer-reviewed journal and conference publications, which are not part of this thesis:

- Marija Popović, Joshua Ott, Julius Rückin, and Mykel J. Kochenderfer. Learning-based Methods for Adaptive Informative Path Planning. *Journal on Robotics and Autonomous Systems (RAS)*, 179:104727, 2024.
  doi: 10.1016/j.robot.2024.104727.

- Apoorva Vashisth, Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Deep Reinforcement Learning with Dynamic Graphs for Adaptive Informative Path Planning. *IEEE Robotics and Automation Letters (RA-L)*, 9(9):7747–7754, 2024. doi: 10.1109/LRA.2024.3421188.

- Jonas Westheider, Julius Rückin, and Marija Popović. Multi-UAV Adaptive Path Planning Using Deep Reinforcement Learning. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023. doi: 10.1109/IROS55552.2023.10342516.

- Gianmarco Roggiolani, Julius Rückin, Marija Popović, Jens Behley, and Cyrill Stachniss. Unsupervised Semantic Label Generation in Agricultural Fields. In *Frontiers in Robotics and AI*, 12, 2025. doi: 10.1109/IROS55552.2023.10342532.

- Alireza Ahmadi, Julius Rückin, Michael Halstead, Marija Popović, and Chris McCool. OptimWeeder: A Reinforcement Learning-Based Approach to Control Mobile Multi-Axes Weeding Systems. *Computers and Electronics in Agriculture, under review*, 2024

- Tobias Zaenker, Julius Rückin, Rohit Menon, Marija Popović, and Maren Bennewitz. Graph-based View Motion Planning for Fruit Detection. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023. doi: 10.1109/IROS55552.2023.10342532.

- Liren Jin, Xieyuanli Chen, Julius Rückin, and Marija Popović. NeU-NBV: Next Best View Planning Using Uncertainty Estimation in Image-Based Neural Rendering. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023. doi: 10.1109/IROS55552.2023.10342226.

- Liren Jin, Julius Rückin, Stefan H. Kiss, Teresa Vidal-Calleja, and Marija Popović. Adaptive-Resolution Field Mapping Using Gaussian Process Fusion with Integral Kernels. *IEEE Robotics and Automation Letters (RA-L)*, 7(3):7471–7478, 2022. doi: 10.1109/LRA.2022.3183797.

Additionally, we made our RL-based adaptive IPP approaches, the adaptive IPP framework for active learning, and the semi-supervised learning method for active learning publicly available to encourage further research:

- Chap. 4 presents our RL-based adaptive IPP algorithm for information gathering in continuous-valued terrain monitoring missions. The implementation is available online at: `https://github.com/dmar-bonn/ipp-rl`

- Chap. 5 presents a map-agnostic formulation of the adaptive IPP problem for a broad family of terrain monitoring missions, enabling training of more generally applicable adaptive IPP strategies. The implementation is available online at: `https://github.com/dmar-bonn/ipp-rl-gen`
- Chap. 6 presents our IPP framework for active learning of semantic robotic vision. The IPP framework implementation is available online at: `https://github.com/dmar-bonn/ipp-al-framework`. The uncertainty-aware Bayesian semantic segmentation network proposed as part of this framework is available online at: `https://github.com/dmar-bonn/bayesian_erfnet`
- Chap. 7 presents our semi-supervised learning approach for active learning of semantic robotic vision using adaptive IPP. The implementation is available online at: `https://github.com/dmar-bonn/ipp-ssl`

Chapter 2

Related Work

There has been remarkable progress in the field of robotics, driven by the ever-growing need to automate complex, repetitive, costly, dangerous or error-prone tasks in various domains [7, 36, 56]. In recent years, tremendous advances in deep learning-based computer vision have enabled robots to perceive and interpret environments in which they are deployed [59, 81, 168], leveraging advanced sensors, such as RGB-D cameras [43, 55, 151] and LiDAR scanners [85, 110, 188]. To achieve autonomous execution of tasks, robots need to plan their actions to reach the task's goal based on perceived environment information. Various methods for local navigation and planning of paths in an a priori known environment exist [73, 82, 87]. However, planning robot actions in an initially unknown environment to maximise the robot's understanding of the environment under limited budget constraints is still an open and challenging research question.

To achieve autonomy, robots must operate in and efficiently gather information about unknown environments. As a solution to this problem, IPP methods aim to plan robot actions that maximise the gathered information using noisy onboard sensors considering resource constraints, such as time or energy [61, 63, 102, 106]. IPP was applied in information gathering tasks, such as terrain monitoring [61, 63, 126, 161, 171], exploration [16, 18, 23, 99], and search and rescue [105, 116] using mobile ground [16, 18], aerial [126, 171] or aquatic robots [61].

We divide this chapter into two subsections. To begin, we review IPP approaches for planning informative next actions and discuss the advantages of our RL-based approaches over existing approaches in Sec. 2.1. In Sec. 2.2, we examine the active and semi-supervised learning paradigms known from computer vision research and discuss how our adaptive IPP approaches extend this idea to robotic information gathering. For an in-depth review and taxonomy, we refer the reader to our survey of learning-based adaptive IPP approaches [127].

2.1 Informative Path Planning

Informative path planning methods seek action sequences that maximise the information gathered about an initially unknown environment using noisy onboard sensors while taking into account the robot's limited resources, e.g. flight time or energy [102, 106, 127]. These approaches are applied in various domains, such as terrain monitoring [61, 63, 126, 161, 171], precision agriculture [107, 124, 171, 187], exploration [16, 18, 23, 99], or search-and-rescue [105, 116], using diverse robot platforms, such as UAVs [126, 171], unmanned ground vehicles [16, 18], or unmanned surface vehicles [61]. The methods developed in this thesis are mostly concerned with terrain monitoring and exploration tasks deployed on UAVs. Our methods could be applied to other robot platforms, e.g. unmanned ground vehicles and unmanned surface vehicles, with minimal platform- and sensor-dependent changes, and could be extended to other environments, e.g. in indoor scenarios.

IPP approaches can be categorised into non-adaptive and adaptive planning methods. Non-adaptive approaches pre-plan static paths prior to a mission [48, 106, 163] while adaptive methods replan paths onboard based on the robot's current understanding of the environment [30, 61, 63, 102, 117]. Many non-adaptive methods pre-compute paths that exhaustively and often uniformly cover the entire environment following geometrically motivated patterns, such as lawnmower-like coverage paths [28, 108]. These methods assume homogeneous distribution of information, e.g. of areas of interest, or a user being interested in exploring the entire terrain apart from certain areas of interesting. Furthermore, they assume homoscedastic uncertainty [76] in collected sensor measurements, i.e. sensor noise independent of the robot's state, e.g. a UAV's altitude, and the measured observation, e.g. a sensor that detects weeds and crops in an arable field equally well. If one or both assumptions do not hold, these non-adaptive methods plan suboptimal paths regarding information-gathering efficiency.

In practice, users are often interested in non-homogeneously distributed areas of interest, such as clusters of high weed pressure in arable fields. The task of terrain exploration can be viewed as a special case in this more general class of tasks, where areas of interest span the entire terrain. Additionally, most sensors have heteroscedastic uncertainty characteristics [76], e.g. due to lower ground sampling resolution at higher UAV altitudes or due to deep learning-based sensors that degrade in performance on underrepresented classes in long-tailed problems [189]. Adaptive IPP methods were proposed for various tasks and deployed on different robot platforms. Adaptive methods consistently show better information-gathering performance than non-adaptive approaches [30, 61, 63, 102, 117].

Adaptive IPP methods can be categorised into non-learning-based [30, 61, 63, 102, 117] and learning-based planning approaches [19, 24, 29, 32, 116, 174].

Many adaptive IPP approaches enhance their planning performance by using some form of learning to update environment maps based on previously collected measurements, e.g. using Gaussian processes [61, 63, 102, 117, 117, 126], or completing unknown space using neural networks [35, 50, 94, 150, 164, 192]. Here, we categorise adaptive IPP methods as non-learning-based planning methods if they solely employ learning-based techniques to update or enhance mapped sensor information but do not use learning-based methods to perform path planning. The learning-based adaptive IPP methods proposed in this thesis also employ Gaussian process-based mapping [61, 63, 102, 126]. Such learning-based mapping methods can potentially enhance the robot's understanding of the environment and subsequently better inform the decision-making in the planning algorithm. We view these methods as an interesting orthogonal research direction to learning-based planning methods and do not include them in the subsequent discussion. We refer the reader to Sec. 4.1 of our survey [127] for more details.

Various non-learning-based adaptive IPP approaches have been proposed. These methods mainly differ in their replanning procedures and can be categorised into sampling-based, optimisation-based, and geometric planning algorithms. Sampling-based methods iteratively sample potential paths and evaluate the sampled paths' information values based on the robot's current understanding of the environment [11, 30, 63, 117, 142]. These approaches build upon well-known sampling-based search algorithms, such as receding horizon planning [63, 142] or Monte Carlo tree search [11, 30, 117], to gradually steer the sampling process towards paths of higher information value. Receding horizon-based algorithms plan in continuous robot workspaces while Monte Carlo tree search-based algorithms usually assume a finite number of discrete actions the robot can execute. Due to the exponential growth of candidate paths with increasing non-myopic planning horizons, non-learning-based sampling-based IPP algorithms require sampling many potential future paths to generate high-quality planning solutions.

In contrast to sampling-based methods that iteratively sample waypoints to find a path, optimisation-based methods aim to directly optimise the information value of a complete path in the space of all possible paths the robot can execute. Optimisation-based methods differ in the algorithms they use to solve this optimisation problem. Most works utilise derivative-free blackbox optimisation methods, such as variants of evolutionary algorithms [61, 126] or Bayesian optimisation algorithms [102, 171], to plan paths in continuous space. Similar to sampling-based methods, blackbox optimisation methods often require many potential candidate paths for which evaluating their information value is compute-expensive, resulting in slow replanning of paths. Ott et al. [117] propose a value-based dynamic programming algorithm to optimise informative paths over a discrete grid of waypoints. Although they propose a fast-to-compute information value of po-

tential waypoints along a path, dynamic programming approaches suffer from the exponential growth of paths to evaluate. In contrast to the planning problems considered in this thesis, they assume knowledge of a robot's goal destination, which allows for an efficient decomposition of the problem. Other approaches aim to compute the gradient of the information value function with respect to the continuous-time path and then leverage gradient-based first- or second-order optimisation algorithms [37, 111, 118]. These approaches require fully differentiable information value functions, robot motion and sensor models, making them less flexibly applicable to various information-gathering setups.

Geometry-based methods collect potential next robot poses based on the geometry of the already explored space [138, 141, 182]. Commonly, these methods select candidate poses at frontiers of explored and unexplored environment spaces. The information value of each candidate pose is evaluated, and the pose that holds the maximal information value is greedily selected [136, 138, 141, 182]. These methods have been proven to be efficient for the exploration of entire environments as frontiers lead to paths that aim to maximise the explored area [182] while possibly considering additional criteria, such as sensor uncertainty [136, 138]. In scenarios where information about certain areas of interest should be precisely mapped, e.g. hotspots of high surface temperature, frontier-based methods struggle to inspect these areas more closely as soon as they are discovered once.

Non-learning-based adaptive IPP methods show promising results, consistently outperforming classically used non-adaptive path planning methods in robotic information gathering. However, non-learning-based methods tend to be computationally inefficient in replanning paths based on the evolving understanding of the environment [19, 126, 135]. The information criteria estimating a path's information value are commonly expensive to compute. As many potential future paths must be evaluated to find informative paths, non-learning-based methods often do not allow for fast or frequent online replanning. Reducing the number of candidate paths these methods evaluate leads to faster replanning but often degrades information-gathering performance during deployment. This makes it challenging to deploy them on resource-constraint robots.

In recent years, learning-based methods have been proposed to tackle the adaptive IPP problem, providing higher compute efficiency and achieving similar or better planning performance on some information-gathering tasks. All learning-based methods achieve this by shifting the computational burden of finding a well-performing planning policy to an offline training phase and inferring the learned planning policy at deployment [4, 19, 24, 29, 32, 83, 98, 116, 130, 169, 174]. These approaches can be categorised into imitation learning methods [4, 29, 32, 98, 130] and RL methods [19, 24, 83, 116, 169, 174] used to learn planning policies. Imitation learning methods learn a policy in a supervised fashion that maximises

the likelihood of resembling an expert adaptive IPP policy [123]. These expert policies could be generated by human operators guiding the robot manually or using previously discussed non-learning-based adaptive IPP methods. In contrast, RL methods learn policies in a trial-and-error interaction with an environment to maximise the experienced sum of rewards received along executed paths.

In this thesis, we focus on RL-based methods to learn adaptive replanning policies since imitation learning is often limited by the quality and quantity of the expert policy data available. Furthermore, we exploit probabilistic environment map representations capturing the robot's understanding of the environment to construct information-theoretic rewards that incentivise the robot to minimise the remaining map uncertainty. These rewards, so-called intrinsic rewards [26], can be computed in an unsupervised fashion without prior knowledge of expert data or an environment in which the robot is trained.

RL-based methods have been proposed for specific adaptive IPP tasks, such as environment exploration or terrain monitoring of areas of interest using occupancy grid maps [18, 22, 98, 116, 169] or monitoring interesting continuous physical phenomena using Gaussian processes [19, 27, 34, 174]. These works mainly differ in the design of their reward function influenced by the mission goal, robot and environment state representation, policy and critic function approximators, and actions space design. Methods for exploration of the entire environment design reward functions measuring coverage of the environment [14, 83, 98, 192], while methods considering precisely mapping areas of interest on a terrain commonly reward decreasing map uncertainty in these areas [3, 19, 34, 170, 185]. All works store a spatial map in their state representation using different map representations, such as occupancy maps [22, 23, 98, 100, 116, 184, 192], sub-sampled Gaussian processes [27, 34, 169, 174], or topological graph maps [18, 19, 24, 185]. Most works use well-known RL algorithms to train planning policies, e.g. classical off-policy deep Q-network [22, 27], sample-efficient off-policy soft actor-critic [18], or popular on-policy proximal policy optimisation [18, 19, 100, 184]. In Chap. 4, we use RL to train policies and value functions to steer the sampling process and to evaluate future paths' information values. In this way, we circumvent sample-inefficient candidate path selection and compute-expensive information value evaluation. Our RL-based approaches not presented in this thesis learn multi-robot adaptive IPP policies for terrain monitoring [175] and single-robot policies for viewpoint planning in 3D environments [167], e.g. in orchards.

In contrast to non-learning-based adaptive IPP methods, these prior RL policies for adaptive IPP, including our approach proposed in Chap. 4, are limited in their general applicability to varying information-gathering tasks. These RL-based approaches design policies tailored towards one specific information-gathering task, assuming occupancy maps to map discrete-valued terrain features,

such as arable field or urban area classification, or pre-trained Gaussian processes to map continuous-valued terrain features, such as surface temperature or signal strengths. These methods use their map representations directly in the planning state representation, thus requiring adaptation and re-training as the map, and hence, planning state representations change. This prohibits the application of learned policies to various monitoring missions that require different map representations. Moreover, these works consider a static user-defined criterion for areas to qualify as interesting, e.g. a fixed temperature threshold or semantic class. Thus, these approaches tend to overfit a static user-defined criterion for areas of interest, dropping in performance as user interests change.

In Chap. 5 of this thesis, we propose a novel map-agnostic formulation of the adaptive IPP problem for terrain monitoring. Our new planning state representation unifies varying map representations and user-defined criteria qualifying areas as interesting. Combining this map-agnostic state space with a new reward function, we train a single adaptive IPP policy using RL that is applicable to continuous- and discrete-valued terrain feature monitoring missions. In this way, our approach facilitates the deployment of learned adaptive IPP policies without re-designing and re-training approaches for specific terrain monitoring missions.

2.2 Adaptive Informative Path Planning for Active Learning

Adaptive IPP methods discussed in Sec. 2.1 assume a reliable performance of the onboard sensors used to interpret and extract information from incoming measurements of the unknown environment. However, this assumption often does not hold when using deep learning-based robotic vision systems to extract the information of interest from sensor measurements. As the environment is unknown, the incoming sensor measurements often differ from the ones the vision system was trained on. Hence, the prediction quality of deep learning-based robotic vision degrades, which limits the amount of information extracted from sensor measurements. Thus, time-consuming and costly human labelling of collected images is needed to re-train and improve these vision systems. In response, some approaches aim to collect the most informative images to improve a robot's deep learning-based vision system while reducing human labelling costs. These approaches combine elements of active learning with robotic planning.

The goal of active learning is to maximise model performance while minimising the amount of labelled training data used to train the model. It assumes the existence of a large unlabelled data pool, then iteratively selects a single data point from the pool by maximising an acquisition function until a labelling budget

is exceeded [17, 44, 92, 165]. Settles [148] provides a comprehensive overview of active learning approaches for low-dimensional machine learning problems. Recent active learning approaches focus on training deep learning models from high-dimensional inputs, e.g. images, where a single data point has a negligible effect on model performance. Active learning methods for deep learning collect a batch of data from the unlabelled data pool instead of single data points, called batch-mode active learning [47, 53, 70, 147]. However, these strategies decide which images to label from an existing already collected large data pool. We directly deploy the robot in an unknown environment. Thus, we do not have access to such a data pool before the robot's deployment.

In contrast, our methods in Chap. 6 and Chap. 7 of this thesis propose novel adaptive IPP frameworks for active learning, collecting new batches of to-be-labelled data directly by deploying a robot in an initially unknown environment. We link the active learning acquisition function to a planning objective, adaptively guiding the UAV towards areas of informative training data and show how to incorporate recently proposed active learning acquisition functions [8, 12, 47] into planning algorithms. Furthermore, we investigate how planners, planning objectives, and terrain mapping influence the active learning performance. In the following, we discuss the previously proposed acquisition functions used in the classical active learning setup assuming an existing unlabelled data pool.

2.2.1 Active Learning for Computer Vision

Active learning methods used in computer vision can be categorised into probabilistically motivated uncertainty-based approaches, geometrically motivated representation-based approaches, and hybrid approaches using a combination of uncertainty- and representation-based methods.

Uncertainty-based active learning methods select data with the highest model uncertainty [8, 47, 70, 71]. Early methods use Gaussian processes [71] or support vector machines [70] to quantify model uncertainty in tasks with low-dimensional inputs. Measuring model uncertainty requires integrating model predictions with respect to all model parameters. As deep neural networks have high-dimensional model parameter spaces, it is computationally challenging to compute their model uncertainty. One approach is to estimate the model uncertainty deterministically in a single forward pass. Although computationally efficient, these methods often do not deliver well-calibrated uncertainty estimates for real-world robotic vision tasks [128]. This leads to uncertainty estimates that do not reflect the actual model prediction quality on data collected during deployment. Alternatively, Gal and Ghahramani [46] propose using dropout [159] at test time, running multiple forward passes with different smaller parts of the model parameters selected randomly. This technique is called Monte Carlo dropout and efficiently approxi-

mates the Bayesian posterior over the model parameters. They use Monte Carlo dropout in acquisition functions maximising model uncertainty, applied to image classification [47]. Monte Carlo dropout can be seen as an ensemble of smaller networks selected from a single trained large network. Other works use neural network ensembles for model uncertainty estimation [84, 122] where each network is independently initialised and trained. Despite higher training costs, independently trained ensembles achieve better prediction performance and uncertainty calibration than Monte Carlo dropout [8, 39]. Recent advances make ensemble training computationally more efficient [39, 67]. In Chap. 6, we study the applicability of different uncertainty-based active learning acquisition functions [8, 46] in the context of robotic planning as mission objectives.

Representation-based active learning methods maximise training data diversity by selecting data points with novel representations in the model's learned latent feature space [40, 147, 155]. Approaches inspired by generative adversarial networks use a generator to learn the joint data representation, while the discriminator distinguishes between labelled and unlabelled data [40, 155]. Sener and Savarese [147] select a minimal number of data points, called a core-set, geometrically covering an unlabelled data pool in the model's latent space. However, both approaches require large in-domain data pools to learn rich representations of the data-generating distribution. These methods are impractical in our adaptive IPP scenario as robots operate in unknown and visually varying environments. In contrast, Blum et al. [12] propose a method for quantifying data novelty in semantic segmentation tasks without access to large in-domain data pools. They use kernel-density estimation of unlabelled images in the network's learned latent space, i.e. the learned embedding of an image into a manifold resembling semantic similarity between images the network was trained on. Pixels of an image are considered novel if they are embedded into regions of the latent space different from the ones images used for network training are embedded into. They use this novelty estimation in a local image-based planning objective and apply it for active learning in aerial semantic mapping. We integrate their novelty estimation into our new global map-based planning objectives proposed in Chap. 6. We rigorously analyse its active learning performance using various global planning schemes and datasets. Our experimental results indicate higher active learning performance using uncertainty-based planning objectives instead of the representation-based objective proposed by Blum et al. [12].

Uncertainty-based methods tend not to draw data points from the data-generating distribution but favour rare data points and outliers. In contrast, representation-based methods tend to select non-informative, easy-to-classify data points at later stages with more training data accessible. Hybrid approaches combine uncertainty- and representation-based methods [93, 173, 183, 190]. Combin-

ing both paradigms, especially in batch-mode active learning, has been shown to result in a strong performance. Zhdanov [190] selects data points near k-means cluster centres of the unlabelled data pool to extract representative samples [190]. Li and Guo [93] fit a Gaussian Process on the unlabelled data pool to compute the mutual information between labelled and unlabelled data points as a measure of representativeness. Wang and Ye [173] introduce an alternating optimisation between maximising mean discrepancy of labelled and unlabelled data in the representations' latent space and uncertainty of selected batches. Similar to Blum et al. [12], Yang et al. [183] use the cosine distance in the network's latent space to select representative samples from the most uncertain images in the unlabelled pool. These methods rely on the existence of a large unlabelled data pool. Thus, they are not directly applicable if the environment is initially unknown.

2.2.2 Robotic Planning for Active Learning

Using autonomous robots to reduce manual labelling effort for training deep learning models is a relatively unexplored research area. In the following, we discuss planning methods that improve robotic vision in unknown environments. These approaches can be categorised as self-supervised or fully supervised learning methods, depending on the human labelling required to train the vision model.

Georgakis et al. [51] propose an approach for active semantic goal navigation which uses ensembles to estimate model uncertainty in their planning objective. Other approaches introduce self-supervised methods to improve or adapt the robot's vision to new environments, eliminating the need for manual labelling. Frey et al. [45] introduce a self-improving continual learning framework for semantic segmentation in indoor scenes by generating pseudo labels from 3D maps. Zurbrügg et al. [191] extend this approach to an embodied agent autonomously navigating towards high training data novelty viewpoints. Chaplot et al. [20] suggest a similar self-supervised approach for semantic segmentation in indoor scenes training an exploration policy with RL to target uncertain 3D map parts. The policy training depends on the simulation environment and the current vision network at the same time. As RL performance tends to degrade with simulation to real-world gaps, this method requires realistic domain-specific simulators and introduces policy re-training costs after each vision network re-training.

Although self-supervised planning approaches for active learning do not require any human labels [20, 45, 191], as discussed by Chaplot et al. [20], they rely on large labelled indoor datasets for pre-training a semantic segmentation model to produce high-quality pseudo labels in new indoor scenes [20, 45, 191]. If the pre-trained model misclassifies objects, these errors not only prevent learning semantics, but could even be reinforced in the case of over-confident predictions. Zurbrügg et al. [191] experimentally show that the model improvement

strongly depends on the chosen pre-training dataset and on the indoor environment in which the robot is deployed. Aerial mapping missions present even more visual variability, with few and often small pre-training datasets available, further exacerbating these issues. As the environment and often domains are initially unknown, these purely self-supervised methods require enormous engineering work to relax the above-mentioned assumptions. Hence, self-supervised methods are not directly applicable to our use case. Our experimental analysis shows that self-supervised approaches applied to UAV-based aerial semantic monitoring missions drastically lack active learning performance compared to our fully supervised approach. Further, our results verify that systematic prediction errors of self-supervised methods limit the model performance improvements.

Most similar to our approach in Chap. 6 is the local planning approach of Blum et al. [12] for active learning in semantic mapping. Their planning objective aims to promote training data novelty in semantic prediction tasks. We combine their ideas on novelty estimation for active learning with our adaptive IPP framework for active learning. In contrast to Blum et al., we propose a general unified adaptive IPP framework supporting probabilistic semantic mapping, various acquisition functions, planning objectives, and global map-based planning algorithms. We also provide in-depth empirical analyses and show that our map-based planners outperform the state-of-the-art local planner proposed by Blum et al..

2.2.3 Semi-Supervised Learning of Robotic Vision

Although our fully supervised adaptive IPP framework for active learning shows state-of-the-art active learning performance, it requires dense human-labelled pixel-wise annotations of selected informative images. To further reduce the required human labelling efforts, we propose a novel semi-supervised approach to adaptive IPP for active learning in Chap. 7. In the following, we discuss recent progress in research on efficient human labelling and semi-supervised learning to enhance model training in case only few or sparse human labels are provided.

Shin et al. [149] have recently introduced an efficient label selection paradigm, which selects a sparse set of pixels for human labelling to train semantic segmentation models. They compute the prediction uncertainty of each pixel and select pixels for human labelling among the most uncertain ones. This targeted pixel selection improves prediction performance faster compared to non-targeted random pixel selection. In a study with human annotators, Benenson and Ferrari [9] show that selecting sparse sets of to-be-labelled pixels reduces human labelling efforts compared to dense pixel-wise labels. In contrast to pixel-wise uncertainty-based selection criteria, Xie et al. [181] propose a method that chooses to-be-labelled image regions used to re-train a pre-trained model in scenarios where the target domain deviates from the source domain. They select the image regions with

highly cluttered model predictions to identify potential prediction errors, which they term region impurity. In Chap. 7, we propose a new sparse human label selection strategy inspired by Xie et al. [181] to drastically reduce human labelling efforts. Instead of selecting image regions, we select individual pixels in regions with high impurity. Opposed to Xie et al. [181], we do not greedily choose pixels with the highest region impurity but randomly select pixels from a set of candidate pixels above a region impurity threshold. We find our method to select informative pixels for human labelling while ensuring label diversity, outperforming state-of-the-art sparse human label selection methods in aerial mapping missions.

Semi-supervised semantic segmentation methods build upon a low budget of human-labelled training samples and improve model performance further by generating pseudo labels from model predictions of unlabelled data [60, 120]. Most real-world datasets have long-tailed class distributions, resulting in biased pseudo labels with limited learning signal for model training. To avoid these issues, He et al. [60] propose a method to re-distribute pseudo labels to match the class distribution of labelled data. Conceptionally similar, our semi-supervised training leverages a low number of sparsely human-labelled samples and combines them with automatically generated pseudo labels. In contrast to image-based pseudo label methods for semi-supervised semantic segmentation [60, 120], our robotic planning approach renders pseudo labels from a probabilistic semantic map of the environment. Instead of using dense pixel-wise map-based pseudo labels proposed by Frey et al. [45], we select sparse pseudo-labelled pixels with low map-based model uncertainty. Our pseudo label selection outperforms state-of-the-art methods [60] applied to robotic planning.

Overall, we combine our adaptive IPP method for active learning with our new sparse human label selection and uncertainty-aware pseudo-label generation into a novel semi-supervised robotic planning framework for active learning. This method combines the advantages of self- and fully robotic active learning. We maintain the general applicability and performance of our fully supervised approach while drastically reducing the required human annotations.

Chapter 3

Basic Techniques

The goal of this thesis is to develop adaptive IPP methods that increase the efficiency of robotic information gathering, such as in terrain monitoring missions. Combining approaches that learn decision-making from data and classical non-learning-based robotic planning approaches has the potential to leverage advantages from both paradigms. This could improve performance in robotic information gathering in two ways. First, learning-based methods could increase adaptability towards new, unseen environments a robot might be deployed in. Second, learning-based methods could reduce the classical planning methods' computational requirements, allowing for faster information gathering and facilitating deployment.

In this thesis, we propose new adaptive IPP methods that combine the paradigm of RL with robotic planning. A basic understanding of RL terminology and algorithms is required to understand our learning-based adaptive IPP methods, particularly those presented in Chap. 4 and Chap. 5.

In this chapter, we first introduce the Markov decision process (MDP) framework in Sec. 3.1 to formalise an agent's sequential decision-making in stochastic environments. Next, we state the general RL problem and introduce Bellman equations used in value-based dynamic programming algorithms to solve the RL problem in Sec. 3.2. Finally, in Sec. 3.3, we introduce the family of actor-critic RL algorithms used in this thesis. For an in-depth introduction to dynamic programming, we refer the reader to Bertsekas [10], and for an in-depth introduction to RL algorithms, see Sutton and Barto [162].

3.1 Markov Decision Processes

Reinforcement learning is a branch of machine learning concerned with an agent making sequential decisions. The agent is placed in an environment and interacts with it by executing actions and receiving subsequent observations and rewards

as depicted in Fig. 3.1. An RL algorithm aims to learn optimal sequential decisions that maximise the return, i.e. the sum of rewards over time, through trial-and-error interaction with the environment. Learning optimal sequential decisions through RL constitutes a broad framework encompassing a wide range of decision-making problems with various real-world applications. A non-exhaustive list of decision-making problems that RL has been applied to include learning to play Atari [112, 144] and board games [144, 152, 153], economics and finance [21], and controlling robots, such as legged robots in challenging terrains [62, 88], and drones during autonomous racing [72, 158]. To formalise and unify these sequential decision-making problems, we first introduce the concept of Markov decision processes.

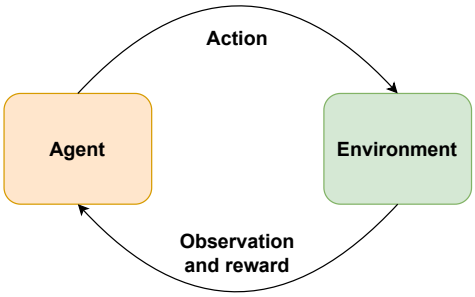


Figure 3.1: Schematic overview of the RL problem. The agent interacts with an environment sequentially by executing actions and then receiving observations and rewards.

A Markov decision process is a tuple $\langle \mathcal{S}, \mathcal{A}, T, R \rangle$ formalising the interaction of an agent with a stochastic environment, where

- $\mathcal{S}$ is the state space;
- $\mathcal{A}$ is the action space;
- $T(s' \mid s, a) \in [0, 1]$ is the state transition function with $a \in \mathcal{A}$ and $s, s' \in \mathcal{S}$;
- $R(s, a, s') \in \mathbb{R}$ is the immediate reward function.

The state space $\mathcal{S}$ is a set of states an environment can be in, and the action space $\mathcal{A}$ is a set of actions an agent can execute in the environment. Both the state and action space can be finite or potentially infinite. The transition function $T(s' \mid s, a)$ is a probability distribution describing the dynamics model of the stochastic environment. Given an environment state s and executed action a, $T(s' \mid s, a)$ is the probability of the environment ending up in a next state s'. Note that the MDP formulation assumes that the environment state transitions only depend on the current state and chosen action, independently from previously chosen actions and states the environment might have been in. This is termed the Markov property. The dynamics model of the environment might or might not be known in advance. RL methods that exploit the existence of or try to learn such a dynamics model from data are called model-based methods, while methods that do not exploit or learn the dynamics model are called model-free methods. The immediate reward function $R(s, a, s')$ rewards or penalises the agent for taking

an action a in a given state s that leads to a next state s' that might be beneficial or detrimental towards achieving a desired goal state $s^g \in \mathcal{S}$. For example, an agent could be rewarded (positive reward function value) for executing an action a in a certain position within a maze that brings the agent closer to the desired goal s^g. In contrast, an agent could also be penalised (negative reward value) for colliding with an obstacle in the environment.

Note that the MDP framework assumes the environment to be fully observable, i.e. we can directly observe and thus know the exact environment states s. However, most real-world decision-making problems, including robotic decision-making problems and the adaptive IPP problem in particular, are not fully observable. Instead, most real-world problems only allow us to partially observe the environment state, e.g. due to a limited sensor range with which an agent can scan the environment or due to uncertain observations from noisy sensor measurements. Thus, partially observable problems are captured by the partially observable Markov decision process framework [79]. In practice, as we also show for our RL-based robotic adaptive IPP methods in this thesis, one can often transform a partially observable Markov decision process into an (approximately) fully observable MDP. This is important as most RL algorithms assume fully observable problems to work well. Based on the MDP formulation derived in this subsection, we subsequently formalise the general RL problem.

3.2 Reinforcement Learning Problem

For a given MDP $\langle \mathcal{S}, \mathcal{A}, T, R \rangle$ as defined in Sec. 3.1, the RL problem aims to learn optimal sequential decisions that maximise the sum of immediate rewards over time. This can be formalised as the following optimization problem

$$\pi^* = \underset{\pi \in \Pi}{\operatorname{argmax}} \underset{\substack{s_{t+1} \sim T(\cdot|s_t, a_t) \\ a_t \sim \pi(\cdot|s_t) \\ s_0 \sim \mu(s)}}{\mathbb{E}} \left[\sum_{t=0}^{N-1} \gamma^t R(s_t, a_t, s_{t+1}) \right], \tag{3.1}$$

where $\pi : \mathcal{S} \to \mathcal{A}$ is a policy mapping states to actions, Π is the set of all possible policy functions, and $\gamma \in [0, 1]$ is a discount factor weighing the importance of future rewards. The policy π can be a deterministic function mapping a state to a deterministic action or stochastic, mapping from a given state to a probability distribution over the action space $\mathcal{A}$.

In this thesis, we are mostly concerned with learning stochastic policies as they allow for applying the large class of policy gradient-based RL algorithms introduced in Sec. 3.3. Additionally, sampling an agent's next action from a stochastic distribution naturally leads to exploration of the environment during policy training, which is crucial for the success of the RL trial-and-error paradigm.

Furthermore, this thesis focuses on episodic instead of infinite-horizon RL problems with a finite number of time steps $N \in \mathbb{N}$ before an episode terminates, e.g. due to energy or time constraints of a robot. Thus, in Eq. (3.1), we draw initial states $s_0 \sim \mu(s_0)$ from a distribution $\mu(\cdot)$ over the state space, where $\tau = (s_0, a_0, \ldots, a_{N-1}, s_N)$ defines an episode occurring with probability

$$p(\tau \mid \pi) = \mu(s_0)\, \pi(a_0 \mid s_0)\, T(s_1 \mid s_0, a_0) \cdots \pi(a_{N-1} \mid s_{N-1})\, T(s_N \mid s_{N-1}, a_{N-1}), \quad (3.2)$$

following a policy π and dynamics model T during the episode. As follows, the notation of the RL problem in Eq. (3.1) can be simplified to

$$\pi^* = \underset{\pi \in \Pi}{\operatorname{argmax}} \mathop{\mathbb{E}}_{\tau}[R(\tau) \mid \pi], \quad (3.3)$$

where $R(\tau) = \sum_{t=0}^{N-1} \gamma^t R(s_t, a_t, s_{t+1})$ is the sum of immediate rewards as in Eq. (3.1) and the expectation over episodes τ is defined according to Eq. (3.2).

Many RL algorithms build on top of the concept of state-value functions or action-value functions to derive algorithms finding optimal policies π^* satisfying Eq. (3.3). The state-value function $V_\pi : \mathcal{S} \to \mathbb{R}$ computes the expected sum of immediate rewards received when starting from a state $s_t = s$ and choosing subsequent actions according to a given policy π as

$$V_\pi(s) = \mathop{\mathbb{E}}_{\tau} \left[\sum_{k=0}^{N-t-1} \gamma^k R(s_{t+k}, a_{t+k}, s_{t+k+1}) \,\middle|\, s_t = s \right]. \quad (3.4)$$

Intuitively, the state-value function evaluates the decision-making quality following a given policy π. Instead, the action-value function $Q_\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ computes the expected sum of immediate rewards received when starting from a state $s_t = s$, choosing a given action $a_t = a$ in this state s_t, and choosing subsequent actions from a_{t+1} on according to the policy π as

$$Q_\pi(s, a) = \mathop{\mathbb{E}}_{\tau} \left[\sum_{k=0}^{N-t-1} \gamma^k R(s_{t+k}, a_{t+k}, s_{t+k+1}) \,\middle|\, s_t = s, a_t = a \right]. \quad (3.5)$$

Intuitively, the action-value function evaluates the benefit of executing a specific action (not necessarily chosen by following policy π) at the current time step t and then following policy π for subsequent time steps $t + k$ for $k > 0$.

Note that the true value functions V_π and Q_π for a given policy π are unknown. Deriving V_π and Q_π for a given policy is called policy evaluation. To this end, we recursively re-define the value functions introduced in Eq. (3.4) and Eq. (3.5) as

$$V_\pi(s) = \mathbb{E}_\tau[R(s_t, a_t, s_{t+1}) + \gamma V_\pi(s_{t+1}) \mid s_t = s], \quad (3.6)$$

$$Q_\pi(s, a) = \mathbb{E}_\tau[R(s_t, a_t, s_{t+1}) + \gamma V_\pi(s_{t+1}) \mid s_t = s, a_t = a], \quad (3.7)$$

Algorithm 1 Model-based Policy Evaluation

Initialise value function $V(s)$ $\forall s \in \mathcal{S}$
while $V(s)$ not converged $\forall s \in \mathcal{S}$ **do**
  for $s \in \mathcal{S}$ **do**
    $V(s) = \sum_{a \in \mathcal{A}} \pi(a \mid s) \sum_{s' \in \mathcal{S}} T(s' \mid s, a)(R(s, a, s') + \gamma V(s'))$ (Eq. (3.6))
return value function V as solution to V_π

using the linearity of the expectation operator, the Markov property, and the definition of the state-value function in Eq. (3.4). Eq. (3.6) and Eq. (3.7) are also known as the Bellman expectation equations, which allow us to iteratively find V_π and Q_π in a model-based fashion using immediate rewards $R(s_t, a_t, s_{t+1})$ and the dynamics model $T(s_{t+1} \mid s_t, a_t)$ as shown in Alg. 1. If the dynamics model is unknown, we can sample the expectations in Eq. (3.6) and Eq. (3.7) using Monte Carlo estimates of the agent's interactions with the environment according to its policy π. Model-free policy evaluation is further discussed in the Sec. 3.3.

Furthermore, we reformulate the RL problem of finding an optimal policy π^* satisfying Eq. (3.3) as the problem of finding the optimal state-value function

$$V^*(s) = \max_{\pi \in \Pi} V_\pi(s) = V_{\pi^*}(s), \tag{3.8}$$

or finding the optimal action-value function

$$Q^*(s, a) = \max_{\pi \in \Pi} Q_\pi(s, a) = Q_{\pi^*}(s, a), \tag{3.9}$$

by definition of Eq. (3.3), the state-value function of a given policy π^* in Eq. (3.4), and the action-value function of a given policy π^* in Eq. (3.5). This established equivalence between optimal value functions V^* or Q^* and optimal policies π^* leads to a family of value-based algorithms that solve the RL problem in Eq. (3.3) by finding the optimal value functions V^* or Q^* instead of finding the optimal policy π^* in the space of policy functions Π.

Value-based algorithms use variants of the classical value iteration formalised in Alg. 2. By repeatedly applying the Bellman optimality equations in Line 4 to an initial guess of the state-value function V, V provably converges to the optimal state-value function V^* [10]. We can derive an optimal policy π^* from the optimal state-value function V^* given the dynamics model T as in Line 7.

The most popular successor of value iteration is Q-learning, which aims to learn the optimal action-value function Q^* by repeatedly applying Bellman optimality equations and using function approximators to represent Q^*, e.g. neural networks [112]. The optimal policy can be derived by $\pi^*(s) = \max_{a \in \mathcal{A}} Q^*(s, a)$, i.e. by inputting the current state s into the function approximator of Q^* and choosing action a with the highest Q-value $Q^*(s, a)$. Thus, Q-learning is a model-

Algorithm 2 Model-based Value Iteration

Initialise value function $V(s)$ for all $s \in \mathcal{S}$
while $V(s)$ not converged for all $s \in \mathcal{S}$ **do**
  for $s \in \mathcal{S}$ **do**
    $V(s) = \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} T(s' \mid s, a)(R(s, a, s') + \gamma V(s'))$
Initialise policy π for all $s \in \mathcal{S}$
for $s \in \mathcal{S}$ **do**
  $\pi(s) = \text{argmax}_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} T(s' \mid s, a)(R(s, a, s') + \gamma V(s'))$
return policy π as optimal policy

free algorithm that does not require knowledge about the dynamics model T to infer the optimal policy from the optimal action-value function.

In this thesis, we mainly use a different family of RL algorithms called actor-critic algorithms. In contrast to value-based methods, actor-critic methods directly optimise the policy π^* in the space of all possible policies Π. To increase the efficiency of this direct policy search, given a current policy π, actor-critic methods utilise value functions derived from policy evaluation as introduced in this subsection to guide the optimisation process [162]. The following subsection develops a basic understanding of these actor-critic RL algorithms.

3.3 Actor-Critic Algorithms

We introduce a family of RL algorithms called actor-critic algorithms that we use and expand on in the methods proposed in this thesis. In contrast to value-based RL algorithms introduced in Sec. 3.2, actor-critic methods directly optimise a policy in the space of all possible policies Π. Thus, these methods view the RL problem formalised in Eq. (3.3) as a direct optimisation problem and aim to find the optimal policy π^* as the solution to Eq. (3.3) directly.

To manipulate the policy in the space of all policy functions Π, commonly, the policy function $\pi : \mathcal{S} \to \mathcal{A}$ is turned into a function approximator $\pi_\theta : \mathcal{S} \times \Theta \to \mathcal{A}$, where $\theta \in \Theta = \mathbb{R}^D$ is a set of D parameters defining the function π_θ. This way, our optimisation problem in Eq. (3.3) is restricted to the space of all possible policy functions parameterised by some $\theta \in \Theta$. Typically chosen parametric forms range from the space of linear models to neural networks parameterised by θ. Although, in theory, $\Theta \subset \Pi$ and thus it might be that the optimal policy $\pi^* \notin \Theta$, in practice, we choose Θ to be an expressive and large parametric function space that includes (near-)optimal policies π^*. Hence, we can transform the RL

problem in Eq. (3.3) into the following optimisation problem

$$\pi^* = \underset{\pi \in \Pi}{\operatorname{argmax}} \underset{\tau}{\mathbb{E}}[R(\tau) \mid \pi] = \underset{\theta \in \mathbb{R}^D}{\operatorname{argmax}} \underset{\tau}{\mathbb{E}}[R(\tau) \mid \pi_\theta]. \tag{3.10}$$

This optimisation problem can be solved with any derivative-free black box or numerical optimisation algorithm. However, in practice, gradient-based optimisation is often more efficient. To apply gradient-based optimisation methods, such as gradient descent or (quasi-)Newton methods, the gradient of the expectation $\mathbb{E}_\tau[R(\tau) \mid \pi_\theta]$ with respect to θ needs to be computed. This gradient is called the policy gradient. Intuitively, policy gradient RL algorithms aim to estimate this gradient pointing towards the direction in policy space Θ that results in policies with a higher expected sum of immediate rewards. Using the policy gradient theorem [162], the gradient of the expected return can be rewritten as

$$\nabla_\theta \underset{\tau}{\mathbb{E}}[R(\tau) \mid \pi_\theta] = \underset{\tau}{\mathbb{E}}[\nabla_\theta \log\left(p(\tau \mid \pi_\theta)\right) R(\tau) \mid \pi_\theta] \tag{3.11}$$

$$= \underset{\tau}{\mathbb{E}}\left[\sum_{t=0}^{N-1} \nabla_\theta \log\left(\pi_\theta(a_t \mid s_t)\right) R(\tau_{t:N}) \,\middle|\, \pi_\theta\right] \tag{3.12}$$

$$= \underset{\tau}{\mathbb{E}}\left[\sum_{t=0}^{N-1} \nabla_\theta \log\left(\pi_\theta(a_t \mid s_t)\right) Q_{\pi_\theta}(s_t, a_t) \,\middle|\, \pi_\theta\right]. \tag{3.13}$$

In the first step, Eq. (3.11), we use the score function gradient estimator [113] known from maximum (log-)likelihood estimation in statistics and classical supervised learning to transform the gradient of the expectation into an expectation of the gradient of the episodes' τ log-probabilities. In the second step, Eq. (3.12), by definition of Eq. (3.2), the log operator, and linearity of the gradient operator, the product of probabilities turns into a sum of gradients of log-probabilities. In this sum, terms related to the initial state distribution $\mu(s_0)$ and dynamics model $T(s_{t+1} \mid s_t, a_t)$ are dropped as they do not depend on θ. This is crucial as it allows us to derive policy-gradient algorithms model-freely without prior knowledge about the environment's dynamics model T. Instead, we sample an agent's experienced state transitions and immediate rewards while interacting with the environment. We use this experience to compute Monte Carlo estimates $R(\tau_{t:N})$ of the returns of an episode τ from time step t on as an estimate of the unknown true action-value function $Q_{\pi_\theta}(s_t, a_t)$ in Eq. (3.13). This procedure used to estimate the policy gradients based on sampled experience, then updating the policy π_θ in direction of this gradient to improve the current policy, and repeating until convergence, yields the well-known REINFORCE algorithm [178].

Although conceptually simple, Monte Carlo estimates of policy gradients have high variance, making learning sample-inefficient. Actor-critic methods introduce two key ideas to reduce the variance of the gradient estimates, thus improving

learning efficiency and training stability. First, they learn value function approximators, e.g. $Q_\psi(s_t, a_t) \approx Q_{\pi_\theta}(s_t, a_t)$ where Q_ψ could be any model parameterised by $\psi \in \mathbb{R}^{D'}$, ranging from a linear model to a neural network. The variance is reduced by learning approximations of $Q_{\pi_\theta}(s_t, a_t)$ not only for sampled states s_t but also for related states that share similar features. This changes the policy gradient computation formalised in Eq. (3.13) to

$$\nabla_\theta \mathop{\mathbb{E}}_{\tau}[R(\tau) \mid \pi_\theta] = \mathop{\mathbb{E}}_{\tau}\left[\sum_{t=0}^{N-1} \nabla_\theta \log\left(\pi_\theta(a_t \mid s_t)\right) Q_\psi(s_t, a_t) \,\middle|\, \pi_\theta\right]. \tag{3.14}$$

The function approximator $Q_\psi(s_t, a_t)$ is called the critic as it evaluates the current policy π_θ but does not take active decisions on what actions to take next as this is decided by policy π_θ, also called the actor.

A second commonly used technique to reduce the variance of the policy gradient estimates in actor-critic algorithms is reducing the variance of returns $R(\tau)$ using baselines $b : \mathcal{S} \to \mathbb{R}$ that only depend on the state. This changes the policy gradient formalised in Eq. (3.13) to

$$\nabla_\theta \mathop{\mathbb{E}}_{\tau}[R(\tau) \mid \pi_\theta] = \mathop{\mathbb{E}}_{\tau}\left[\sum_{t=0}^{N-1} \nabla_\theta \log\left(\pi_\theta(a_t \mid s_t)\right) \left(Q_{\pi_\theta}(s_t, a_t) - b(s_t)\right) \,\middle|\, \pi_\theta\right]. \tag{3.15}$$

Since $b(s_t)$ does not depend on an action $a_t \sim \pi_\theta(a_t \mid s_t)$, it does not change the policy gradient with respect to θ. It can be shown that the state-value function $b(s_t) = V_{\pi_\theta}(s_t)$ from Eq. (3.4) maximally reduces the variance of the policy gradient estimator in Eq. (3.15) [162]. Intuitively, this critic term tells how much better or worse it is to pick an action a_t than following the policy π_θ in state s_t. We call $A_{\pi_\theta}(s_t, a_t) = Q_{\pi_\theta}(s_t, a_t) - V_{\pi_\theta}(s_t)$ the advantage function of a given policy π_θ. Combining the idea of learned function approximators as critics with the idea of state-value functions as baselines leads us to the following lower-variance policy gradient estimator

$$\nabla_\theta \mathop{\mathbb{E}}_{\tau}[R(\tau) \mid \pi_\theta] = \mathop{\mathbb{E}}_{\tau}\left[\sum_{t=0}^{N-1} \nabla_\theta \log\left(\pi_\theta(a_t \mid s_t)\right) \left(Q_{\pi_\theta}(s_t, a_t) - V_{\pi_\theta}(s_t)\right) \,\middle|\, \pi_\theta\right] \tag{3.16}$$

$$= \mathop{\mathbb{E}}_{\tau}\left[\sum_{t=0}^{N-1} \nabla_\theta \log\left(\pi_\theta(a_t \mid s_t)\right) \left(R(s_t, a_t, s_{t+1}) + \gamma V_{\pi_\theta}(s_{t+1}) - V_{\pi_\theta}(s_t)\right) \,\middle|\, \pi_\theta\right] \tag{3.17}$$

$$\approx \mathop{\mathbb{E}}_{\tau}\left[\sum_{t=0}^{N-1} \nabla_\theta \log\left(\pi_\theta(a_t \mid s_t)\right) \left(R(s_t, a_t, s_{t+1}) + \gamma V_\psi(s_{t+1}) - V_\psi(s_t)\right) \,\middle|\, \pi_\theta\right], \tag{3.18}$$

where Eq. (3.16) holds as the state-value function V_{π_θ} is not changing the gradient since it only depends on the state s_t, and Eq. (3.17) holds by definition of the action-value function in Eq. (3.7). Last, in Eq. (3.18),we make use of a state-value function approximator $V_\psi \approx V_{\pi_\theta}$ parameterised by ψ to approximate the true

Algorithm 3 "Vanilla" Actor-Critic Algorithm

Initialise actor and critic parameters θ, ψ
while policy π_θ not converged **do**
  Collect episodes τ following policy $a_t \sim \pi_\theta(\cdot \mid s_t)$
  Compute policy gradient estimate of $\nabla_\theta \mathbb{E}_\tau[R(\tau) \mid \pi_\theta]$ with Eq. (3.18)
  Update actor $\theta \leftarrow \theta + \alpha \nabla_\theta \mathbb{E}_\tau[R(\tau) \mid \pi_\theta]$ via gradient ascent
  Fit critic V_ψ optimising Eq. (3.19) using gradient descent
return policy π_θ as (approximately) optimal policy

advantage function $A_{\pi_\theta}(s_t, a_t) \approx A_\psi(s_t, a_t) = R(s_t, a_t, s_{t+1}) + \gamma V_\psi(s_{t+1}) - V_\psi(s_t)$. A simple way to learn $V_\psi \approx V_{\pi_\theta}$ is to use ordinary least squares minimisation of the returns over all sampled episodes τ and time steps $t \leq N$ resulting in

$$\underset{\psi}{\operatorname{argmin}} \underset{\tau,t}{\mathbb{E}}[\|V_\psi(s_t) - R(\tau_{t:N})\|^2], \tag{3.19}$$

which is, in contrast to model-based policy evaluation in Alg. 1, a model-free Monte Carlo estimation method to evaluate policy π_θ. Thus, combined with the model-free estimation of policy gradients in Eq. (3.18), actor-critic methods learn policies from experience without any knowledge of the dynamics model T.

Actor-critic methods are most commonly implemented as shown in Alg. 3. First, we collect episodes τ by interacting with the environment following a current policy π_θ. Second, we estimate the policy gradient in Eq. (3.18) as the sum over all episodes and time step terms $\nabla_\theta \log(\pi_\theta(a_t \mid s_t)) A_\psi(s_t, a_t)$. Third, we use gradient ascent to update policy parameters θ in the direction of the policy gradient estimate. Last, we refit the critic using gradient descent to satisfy Eq. (3.19). We repeat the procedure until convergence of policy π_θ.

Most modern variants of actor-critic RL algorithms are concerned with reducing the variance of the policy gradient estimate when using neural networks as actors and critics, i.e. how to design sample-efficient low-variance baselines without suffering from highly biased gradient estimates. Proximal policy optimisation (PPO) [146] is a state-of-the-art actor-critic algorithm that we also use in this thesis for RL-based adaptive IPP methods. PPO proposes to use a generalised version of the advantage function approximator A_ψ in Eq. (3.18) to increase sample efficiency [145]. Furthermore, PPO restricts its policy updates to gradient ascent step sizes α in Line 5, Alg. 3, which ensure small but stable policy updates. This increases training stability since updates to a policy that are too large make learning critic V_ψ of a given policy π_θ challenging. Apart from these two important modifications, PPO and other state-of-the-art actor-critic algorithms share the same underlying concepts with the vanilla actor-critic algorithm in Alg. 3.

In general, model-free actor-critic algorithms use a provided (often simulated) environment to collect experience executing a current policy. Access to the en-

vironment's dynamics model is not required, making actor-critic methods easily applicable for policy learning in complex environments without well-known dynamics models. Based on the collected experience, the current stochastic policy and critic are updated in a gradient-based fashion. The policy is updated by the policy gradients estimated using the critic which evaluates the policy's performance. To represent policies and critics, we commonly parameterise both functions as neural networks. This way, leveraging recent progress in neural network architectures, we can process complex environment state information commonly occurring in robotic tasks, such as images from the robot's camera, grid maps and topological maps of the environment.

The following two chapters show how RL algorithms can be used to make information gathering via adaptive IPP more efficient. In Chap. 4, we show how a specific adaptive IPP problem can be modelled as a MDP and reformulated as a RL problem, which allows us to speed up sampling-based robotic planning using stochastic policies and state-value functions learned during training in simulation. Hence, we solve this specific adaptive IPP problem during deployment in a computationally efficient way. In Chap. 5, we derive a state space representation and reward function for a broad family of adaptive IPP problems unified in a novel MDP formulation. In contrast to our method proposed in Chap. 4, this allows us to learn a single planning policy applicable to many varying adaptive IPP problems using any RL algorithm introduced in this chapter.

Chapter 4

Adaptive Informative Path Planning Using Deep Reinforcement Learning

Recent years have seen increasing usage of mobile robots in a variety of information-gathering missions, including environmental monitoring [61, 63, 89, 126], exploration [36], and inspection [48]. These mobile robotic systems promise a safe and economical solution for many applications that require precise measurements at scale [38]. Currently, mobile robots are often manually supervised or operated in environments unknown before deployment. Other robotic systems pre-plan paths along which robots collect data. These static pre-planned paths often limit the information-gathering efficiency if the information is non-uniformly distributed as the robot cannot react to newly incoming measurements. To fully exploit their automation potential, mobile robots need to explore unknown environments autonomously and actively plan their next actions onboard. To achieve this, a key challenge for a robotic system is to reason about where to move next to efficiently collect new informative data based on its incomplete understanding of the environment and resource constraints, such as limited onboard energy and compute. In literature, this problem is also known as the adaptive informative path planning problem [106, 127].

This chapter examines the problem of precisely mapping user-defined areas of interest in an unknown environment using a resource-constrained UAV. The UAV is equipped with an onboard camera to collect noisy measurements and is limited in flight time and compute power. The environment is characterised by an a priori unknown non-uniform 2D scalar feature field on the terrain, e.g. surface temperature, humidity or signal strength. A user-defined range of feature values qualifies an area of interest, e.g. indicating hotspots [61, 126]. To maximise the information gathered about the areas of interest, the UAV adaptively

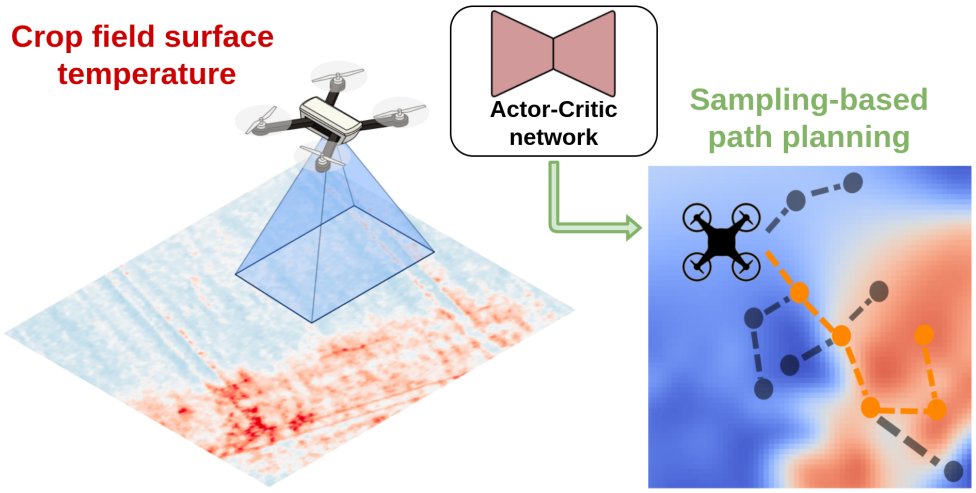


Figure 4.1: Left: Our RL-based adaptive IPP approach applied in a UAV-based crop field surface temperature monitoring mission aiming to precisely map temperature hotspots (red). Right: The UAV's current surface temperate map belief (top-down view of the flat terrain) is induced by previously collected camera measurements. Based on the map belief, the UAV plans a future path (orange) maximising information about hotspots. An actor-critic neural network steers sampling candidate paths (grey) and estimates their information value about hotspots.

replans paths based on its evolving understanding of the environment captured in a continuously updated terrain map.

Many information-gathering approaches tackling the adaptive IPP problem have been proposed [30, 61, 63, 126, 171]. These approaches enable adjusting the robot's path based on incoming sensor measurements and its potentially changing understanding of the environment. Moreover, these works show that adaptive replanning of paths increases the information-gathering efficiency in many applications compared to executing pre-planned static paths. Adaptively replanning paths is commonly realised using sampling-based [30, 63] or optimisation-based [61, 126, 171] planning methods. Both approaches involve an iterative two-step procedure converging to informative future paths. A candidate path is selected or sampled, and its expected information value about areas of interest is estimated. This is done by simulating potential future measurements along the path based on the robot's current terrain map and updating it accordingly. This procedure is computationally expensive for two reasons. First, the number of future candidate paths grows exponentially with the path's length, often requiring many iterations to converge to informative future paths. Second, computing the information value of a path is slow as computing expected future map updates is computationally expensive for commonly chosen scalar field map representations, such as Gaussian processes [61, 102] or Kalman filters [126]. In this chapter, we aim to overcome these limitations leveraging offline-trained neural networks that guide the replanning procedure during deployment in a compute-efficient fashion.

The main contribution of this chapter is a new adaptive IPP approach for UAV-based information gathering illustrated in Fig. 4.1. Our method combines sampling-based planning with reinforcement learning to accelerate onboard replanning of paths during deployment. Specifically, we train a strategy that steers the sampling of candidate paths towards potentially informative future paths and train an information value estimator offline in simulated information-gathering missions using RL. During deployment, we perform sampling-based planning of future paths guided by the offline-learned sampling strategy. Furthermore, the sampled candidate paths' information values are evaluated using the offline-learned information value estimator. In this way, our planning procedure requires fewer iterations to converge to informative future paths and estimates information values of paths more efficiently, resulting in accelerated replanning of paths.

In sum, we make the following claims. First, our RL-based adaptive IPP method accelerates the replanning of paths compared to non-learning-based state-of-the-art methods, resulting in higher information-gathering performance on simulated terrain monitoring missions. Second, our adaptive IPP method outperforms traditionally used pre-computed coverage paths in real-world dataset-based robotic information-gathering missions. Third, we verify the individual components and design choices of our adaptive IPP algorithm in an ablation study.

This chapter incorporates material from the following peer-reviewed conference publication, for which I have been the main contributor:

- Julius Rückin, Liren Jin, and Marija Popović. Adaptive Informative Path Planning Using Deep Reinforcement Learning for UAV-based Active Sensing. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022

4.1 Adaptive IPP as a Reinforcement Learning Problem

We first review the problem of adaptive IPP in the context of our UAV-based 2D scalar feature field monitoring missions in Sec. 4.1.1. In Sec. 4.1.2, we formalise the connection between the adaptive IPP problem and the general RL problem introduced in Chap. 3. Subsequently, this established connection enables us to design our new RL-based adaptive IPP method detailed in Sec. 4.2.

4.1.1 Adaptive IPP for UAV-Based Terrain Monitoring

We consider a UAV with position $\mathbf{p}_t \in \mathbb{R}^3$ at time step t, equipped with a downwards-facing camera. The UAV is deployed in a terrain $\xi \subset \mathbb{R}^2$ that is

assumed to be flat. The terrain is characterised by its unknown temporally static scalar feature field $F : \xi \to \mathbb{R}$, e.g. a surface temperature field. The UAV's mission goal in robotic information gathering is to find areas of interest $\xi_I \subset \xi$ and precisely map the scalar feature field F in these areas ξ_I. An area of interest is defined by an interesting feature value threshold $f_{th} \in \mathbb{R}$, such that any $\mathbf{x} \in \xi_I$ has a feature value $F(\mathbf{x}) \geq f_{th}$. In this way, the user can define hotspots or anomalies the UAV should focus on while mapping the environment.

The UAV collects camera measurements $\mathbf{z}_t \in \mathbb{R}^{W \times H}$ of the terrain at each time step t, where $W \times H$ is the image resolution. The collected images are assumed to be noisy, such that each pixel i follows $\mathbf{z}_{i,t} = F(\mathbf{x}_{i,t}) + \varepsilon$, where $\varepsilon \sim \mathcal{N}\big(0, \nu(\mathbf{p}_t)\big)$ is zero-centered Gaussian noise with altitude-dependent variance $\nu(\mathbf{p}_t) \in \mathbb{R}$ and $\mathbf{x}_{i,t}$ is pixel's i location downwards-projected on the terrain from position $\mathbf{p}_t$.

To map the scalar feature field F, we follow the probabilistic terrain mapping method of Popović et al. [126]. The map is updated in a probabilistic sequential Bayesian fashion. A prior map distribution $p(F)$ is given by a Gaussian Process with mean function $m : \xi \to \mathbb{R}$ and a kernel function $k : \xi \times \xi \to \mathbb{R}$ defining the covariance between two points in the terrain. To avoid computationally expensive Gaussian process updates at time step t, a Kalman filter with mean $\boldsymbol{\mu}_t$ and covariance matrix $\mathbf{P}_t$ is used to represent the map belief $\hat{F}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{P}_t)$.

To construct a fixed-dimensional map state required to compute Kalman filter updates, the terrain ξ is discretised into a set of equidistantly distributed grid cells with centre positions $\mathcal{X} \subset \mathbb{R}^2$. The prior mean $\boldsymbol{\mu}_0$ is initialised by querying the mean function for each grid cell $\mathbf{x} \in \mathcal{X}$ as $\boldsymbol{\mu}_0(\mathbf{x}) = m(\mathbf{x})$. Similarly, the prior covariance matrix $\mathbf{P}_0$ is initialised by querying the kernel for each pair of grid cells $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$ as $\mathbf{P}_0(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}, \mathbf{x}')$. At each time step t, the map belief $\hat{F}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{P}_t)$ is updated based on newly collected measurements $\mathbf{z}_t$ recorded at a UAV position $\mathbf{p}_t$ and downwards-projected to the flat terrain. This is done by applying the Kalman filter update equations [129] to the previous map belief given by $\boldsymbol{\mu}_{t-1}$ and $\mathbf{P}_{t-1}$. For further details on the Kalman filter updates using the camera measurements, we refer to Popović et al. [126].

Our adaptive IPP problem for UAV-based terrain monitoring aims to find a path $\psi^* = (\mathbf{p}_1, \ldots, \mathbf{p}_N)$ that maximises the information gathered about areas of interest ξ_I, where $\mathbf{p}_t \in \mathbb{R}^3$, $t \in \{1, \ldots, N\}$, are UAV positions above the terrain in the three-dimensional workspace. To this end, the path ψ^* maximises an information criterion $I : \Psi \to \mathbb{R}$ over the set of all possible paths Ψ, so that

$$\psi^* = \underset{\psi \in \Psi}{\operatorname{argmax}} \ \mathrm{I}(\psi), \text{ s.t. } C(\psi) \leq B, \tag{4.1}$$

where $\mathrm{C} : \Psi \to \mathbb{R}^+$ maps a path to its associated execution cost, $B \in \mathbb{R}^+$ is the UAV's mission budget limit. The information criterion $I(\psi)$ is computed based on the set of measurements $\mathbf{z}_{1:N}$ collected along path ψ at positions $\mathbf{p}_{1:N}$.

Since we focus on the scenario of monitoring a terrain using a UAV, the costs $C(\psi)$ of executing path $\psi = (\mathbf{p}_1, \ldots, \mathbf{p}_N)$ are the defined by the flight time

$$C(\psi) = \sum_{t=1}^{N} \mathrm{c}(\mathbf{p}_\mathrm{t}, \mathbf{p}_{\mathrm{t+1}}), \tag{4.2}$$

where $\mathbf{p}_t \in \mathbb{R}^3$ is a 3D measurement position above the terrain the image is recorded from. Hence, the function $c : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}^+$ computes the flight time between measurement positions. We approximate the UAV's flight time by assuming constant acceleration and deceleration $\pm u_a$ and maximum velocity u_v.

Similar to previous adaptive IPP works [61, 102, 126], the information criterion I measures the information gathered about areas of interest ξ_I by evaluating the uncertainty of the map belief $\hat{F}_t$ in these areas. As areas of interest ξ_I are unknown, to adaptively plan paths towards these areas as we discover them, we approximate areas of interest at time step t as [61, 126]

$$\hat{\mathcal{X}}_{I,t} = \left\{ \mathbf{x}_i \in \mathcal{X} \mid \boldsymbol{\mu}_{t,i} - \beta \mathbf{P}_{t,i,i} \geq f_{th} \right\}, \tag{4.3}$$

where $\boldsymbol{\mu}_{t,i}$ and $\mathbf{P}_{t,i,i}$ are the mean and variance of grid cell $\mathbf{x}_i$ at time step t, $\beta \in \mathbb{R}^+$ is a user-defined confidence interval width, and $f_{th} \in \mathbb{R}$ is the user-defined interesting feature threshold. In this way, we consider both areas likely of interest based on gathered measurements and areas in which the map is uncertain.

As in prior works [61, 126], we use the reduction of map uncertainty as the measure of information in Eq. (4.1). Map uncertainty is captured by the trace $\mathrm{Tr}(\mathbf{P}_t)$ of its covariance matrix $\mathbf{P}_t$ as this measure is known to encourage A-optimal map estimates [154]. Furthermore, we restrict our information criterion to only consider map uncertainty reduction in areas $\hat{\mathcal{X}}_{I,t}$ that might be of interest. Specifically, our information criterion $I(\psi)$ for a path ψ is defined as

$$I(\psi) = \sum_{t=1}^{N} \frac{\sum_{i=1}^{|\mathcal{X}|} \mathbb{I}_{\mathbf{x}_i \in \hat{\mathcal{X}}_{I,t-1}} (\mathbf{P}_{t-1,i,i} - \mathbf{P}_{t,i,i})}{c(\mathbf{p}_{t-1}, \mathbf{p}_t)}, \tag{4.4}$$

where the nominator expresses the reduction in the covariance trace from the previous to the current time step after collecting a measurement $\mathbf{z}_t$ at position $\mathbf{p}_t$ and updating the map belief accordingly. The indicator function $\mathbb{I}_{\mathbf{x}_i \in \hat{\mathcal{X}}_{I,t}}$ is one if a grid cell $\mathbf{x}_i$ belongs to an area that might be of interest, else it is zero. Hence, the covariance trace reduction is only considered in areas $\hat{\mathcal{X}}_{I,t-1}$ that were believed to be of interest in the previous time step. In this way, we balance exploration and adaptively focus on already discovered areas of interest. We normalise the information gained at position $\mathbf{p}_t$ by the flight time required to reach $\mathbf{p}_t$. This encourages the UAV to efficiently allocate its maximal flight time B.

4.1.2 Connecting Adaptive IPP and Reinforcement Learning

Next, we cast the adaptive IPP problem for terrain monitoring formalised in Sec. 4.1.1 into a RL problem, enabling us to use RL techniques to train the UAV to plan informative paths. We highlight the RL definitions required to establish this connection and refer to Chap. 3 for an in-depth introduction to RL.

The RL problem aims to find a policy $\pi^* : \mathcal{S} \to \mathcal{A}$ in the space of all possible policies Π that maps an environment state $s \in \mathcal{S}$ to an action $a \in \mathcal{A}$, such that

$$\pi^* = \underset{\pi \in \Pi}{\operatorname{argmax}} \mathop{\mathbb{E}}_{\tau} \left[\sum_{t=0}^{N-1} \gamma^t R(s_t, a_t, s_{t+1}) \,\middle|\, \pi \right] = \underset{\pi \in \Pi}{\operatorname{argmax}} \mathop{\mathbb{E}}_{\tau} [R(\tau) \mid \pi], \tag{4.5}$$

where $a_t \sim \pi(s_t)$, $\gamma \in [0, 1]$ is a discount factor weighing the importance of future rewards $R(s_t, a_t, s_{t+1})$, and $R(\tau) = \sum_{t=0}^{N-1} \gamma^t R(s_t, a_t, s_{t+1})$ is the sum of discounted rewards of an episode $\tau = (s_0, a_0, \ldots, a_{N-1}, s_N)$.

In the terrain monitoring scenario described in Sec. 4.1.1, the state s_t at time step t is only partially observable as the scalar feature field F is unknown and can only be estimated using noisy measurements $\mathbf{z}_t$. However, the probabilistic map belief $\hat{F}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{P}_t)$ given by the Kalman filter mean $\boldsymbol{\mu}_t$ and covariance $\mathbf{P}_t$ recovers a Markov decision process as in Sec. 3.1 with belief states [162]

$$s_t = (\boldsymbol{\mu}_t, \mathbf{P}_t, \mathbf{p}_{t-1}, B_t), \tag{4.6}$$

where $\boldsymbol{\mu}_t$ and $\mathbf{P}_t$ define the probabilistic belief over all possible feature fields F, $\mathbf{p}_{t-1}$ is the UAV position after reaching the previously planned position assuming a noise-free localisation system and $B_t \leq B$ is the UAV's remaining flight time.

In our adaptive IPP problem formalised in Eq. (4.1), a planning policy π maps belief states s_t defined in Eq. (4.6) to the next UAV position $\mathbf{p}_t = a_t = \pi(s_t) \in \mathbb{R}^3$ in the workspace above the terrain, such that the path

$$\psi_\pi = (\mathbf{p}_0, \ldots, \mathbf{p}_{N-1}) = (\pi(s_0), \ldots, \pi(s_{N-1})) \tag{4.7}$$

is induced by the planning policy π. Hence, the RL problem in Eq. (4.5) is equivalent to finding an optimal planning policy $\pi^* \in \Pi$, so that

$$\begin{aligned}
\pi^* &= \underset{\pi \in \Pi}{\operatorname{argmax}} \mathop{\mathbb{E}}_{\tau} [R(\tau) \mid \pi] && (4.8)\\
&= \underset{\pi \in \Pi}{\operatorname{argmax}} \mathop{\mathbb{E}}_{\tau} [I(\psi_\pi) \mid \pi] && (4.9)\\
&= \underset{\pi \in \Pi}{\operatorname{argmax}} \mathop{\mathbb{E}}_{\tau} \left[\sum_{t=1}^{N} \frac{\sum_{i=1}^{|\mathcal{X}|} \mathbb{I}_{\mathbf{x}_i \in \hat{\mathcal{X}}_{I,t-1}} (\mathbf{P}_{t-1,i,i} - \mathbf{P}_{t,i,i})}{c(\mathbf{p}_{t-1}, \mathbf{p}_t)} \,\middle|\, \pi \right] && (4.10)\\
&= \underset{\pi \in \Pi}{\operatorname{argmax}} \mathop{\mathbb{E}}_{\tau} \left[\sum_{t=0}^{N-1} \gamma^t R(s_t, a_t, s_{t+1}) \,\middle|\, \pi \right]. && (4.11)
\end{aligned}$$

The first step in Eq. (4.8) starts from the general definition of the RL problem in Eq. (4.5). In the second step, in Eq. (4.9), we specify the sum of discounted rewards $R(\tau)$ as the UAV's mission goal to maximise the information criterion I as stated in the adaptive IPP formalised in Eq. (4.1). Note that the episode $\tau = (s_0, \pi(s_0), \ldots, \pi(s_{N-1}), s_N)$ encompasses the path ψ_π induced by planning policy π as defined in Eq. (4.7). The third step in Eq. (4.10) makes direct use of the definition of a path's ψ_π information value $I(\psi_\pi)$ formalised in Eq. (4.4). The map belief and UAV position required to compute $I(\psi_\pi)$ are captured in the episode τ as belief states s_t defined as in Eq. (4.6). In the fourth step, we establish equivalence between the adaptive IPP problem and RL problem. The equivalence between both problems holds for $\gamma = 1$ and by defining the reward function of the terrain monitoring scenario detailed in Sec. 4.1.1 as

$$R(s_t, a_t, s_{t+1}) = \frac{\sum_{i=1}^{|\mathcal{X}|} \mathbb{I}_{\mathbf{x}_i \in \hat{\mathcal{X}}_{I,t}} (\mathbf{P}_{t,i,i} - \mathbf{P}_{t+1,i,i})}{c(\mathbf{p}_{t-1}, \mathbf{p}_t)}, \tag{4.12}$$

where actions $a_t = \pi(s_t) = \mathbf{p}_t$ are the next planned measurement positions. We can use this established equivalence of the adaptive IPP and RL problem to train adaptive IPP policies $\pi^*(s_t)$ that process our belief states s_t introduced in Eq. (4.6) and map it to next measurement positions that maximise the sum of our adaptive IPP rewards in Eq. (4.12). In practice, these adaptive IPP policies π^* could be trained with any RL algorithm introduced in Chap. 3.

In our RL algorithm subsequently introduced in Sec. 4.2, we perform sampling-based planning combined with RL. We do not directly use a learned policy π^* to output the next measurement position. Instead, we sample potentially informative next measurement positions from a learned stochastic policy π over all possible next measurement positions to guide the planning process in a sample-efficient fashion. Furthermore, sampling-based planners evaluate a sampled path's ψ_π information value $I(\psi_\pi)$ in a compute-expensive fashion. To circumvent this, we leverage the established equivalence between the information criterion and the sum of discounted rewards in Eq. (4.11). Generally, a policy's π state-value function starting in state s at time step t is defined as

$$V_\pi(s) = \mathop{\mathbb{E}}_{\tau} \left[\sum_{k=0}^{N-t-1} \gamma^k R(s_{t+k}, a_{t+k}, s_{t+k+1}) \,\middle|\, s_t = s \right]. \tag{4.13}$$

This state-value function is equivalent to evaluating the information value $I(\psi_{\pi,t})$ of a path $\psi_{\pi,t}$. This can be seen by inserting our reward in Eq. (4.12), $\gamma = 1$ and using Eq. (4.11). The path $\psi_{\pi,t}$ is induced by policy π and starts at some time step t with belief state $s_t = s$, i.e. $\psi_{\pi,t} = (\pi(s_t), \ldots, \pi(s_{N-1}))$ as in Eq. (4.7). In practice, the policy π used to sample future paths and the state-value function $V_\pi(s_t) = I(\psi_{\pi,t})$ evaluating sampled paths' information values

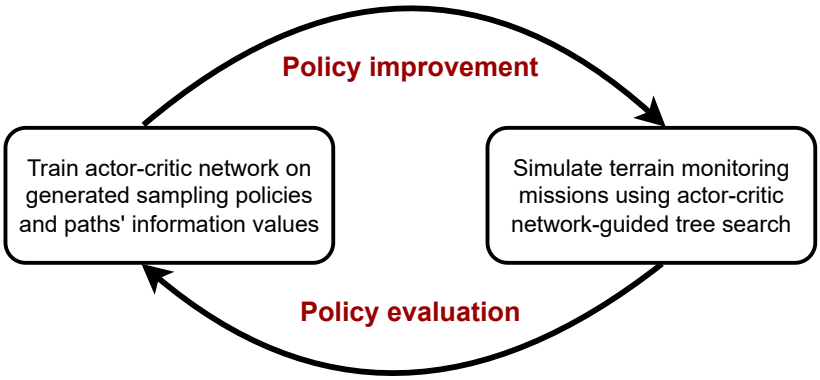


Figure 4.2: Overview of our iterative approach. An actor-critic network predicts informative measurement positions and estimates a path's information value to guide a tree search in simulated information-gathering missions (policy improvement). Then, the neural network is re-trained on the tree search's sampling policy and paths' information values (policy evaluation). This process is iteratively repeated until convergence of information-gathering performance.

can be learned with actor-critic RL algorithms described in Sec. 3.3. Similarly, we describe our new RL-based adaptive IPP method in the following Sec. 4.2 that leverages these established equivalences between RL and adaptive IPP to accelerate sampling-based replanning of informative paths.

4.2 Reinforcement Learning Algorithm

We aim to learn planning policies for adaptive IPP in UAV-based terrain monitoring missions offline in simulation to allow for fast online replanning at deployment. To achieve this, we build upon the connection between adaptive IPP and RL we established in Sec. 4.1. Specifically, we combine recent advances in RL by Silver et al. [152, 153] with sampling-based replanning of paths. Next, Sec. 4.2.1 overviews our RL-based planning policy training procedure.

4.2.1 Algorithm Overview

Our RL-based approach combines Monte Carlo tree search (MCTS) for adaptive IPP [30] with an actor-critic network, conceptually depicted in Fig. 4.2. During training, the algorithm alternates between collecting experience by simulating terrain monitoring missions based on the current actor-critic network and using this experience to re-train and improve the actor-critic network, similarly to classical actor-critic algorithms introduced in Sec. 3.3. Terrain monitoring missions are generated by simulating diverse scenarios with varying 2D scalar feature fields and random initial UAV positions $\mathbf{p}_0$ as explained in Sec. 4.2.2.

At each time step t during a mission, we execute a tree search similar to MCTS that plans the next measurement position $\mathbf{p}_t$ based on the current belief state s_t

introduced in Eq. (4.6) using an actor-critic network as explained in Sec. 4.2.3. The tree search returns a sampling policy $\boldsymbol{\pi}(s_t)$ that mimics the sampling-based search, from which the next measurement position $\mathbf{p}_t$ is drawn. Additionally, the information value $V_\pi(s_t) = I(\psi_{\pi,t})$ of the path induced by π is computed after termination of the mission. The sampling policies and paths' information values are stored in an experience buffer along with respective belief states s_t.

The stored experience is used to re-train the actor-critic network based on the sampling policies and corresponding information values. In Sec. 4.2.4, we introduce the used actor-critic convolutional neural network architecture that processes a belief state. Additionally, we show how to train it based on the collected experience. Finally, we describe implementation details that aim to further improve planning performance and sample efficiency in Sec. 4.2.5.

4.2.2 Mission Generation at Training Time

We aim to learn an actor-critic neural network predicting sampling policies and information values that guide a tree search procedure to efficiently find informative paths during deployment. We train this planning strategy offline on a diverse set of simulated UAV-based terrain monitoring missions.

We randomly generate ground truth 2D scalar fields $F : \xi \rightarrow [0, 1]$ of the to-be-monitored environmental phenomenon with spatial correlations, similarly to Popović et al. [126]. We assume the terrain $\xi = [0, 1]^2$ to be a scale-agnostic unit-square and the feature values $F(\mathbf{x})$ at a point $\mathbf{x} \in \xi$ to be normalised between 0 and 1. This ensures that we can reuse our learned planning strategy for different to-be-monitored continuous scalar features that might vary in absolute magnitudes and units, such as surface temperature or signal strength.

During a mission, downwards-facing camera measurements $\mathbf{z}_{i,t} = F(\mathbf{x}_{i,t}) + \varepsilon$ at UAV position $\mathbf{p}_t$ are simulated independently for each pixel i. Each pixel i is projected onto the flat terrain ξ to find its measurement point $\mathbf{x}_{i,t} \in \xi$. We add altitude-dependent zero-centered Gaussian noise to the ground truth feature field value $F(\mathbf{x}_{i,t})$ by sampling ε from $\mathcal{N}\big(0, \nu(\mathbf{p}_t)\big)$. Following Popović et al. [126], measurement noise $\nu(\mathbf{p}_t)$ increases with higher UAV altitudes as the camera's ground sampling distance increases as well. Hence, the planning algorithm has to balance a larger field of view from higher altitudes, potentially yielding information about larger terrain areas with increasing measurement uncertainty.

We simulate a fixed number of terrain monitoring missions. A mission is terminated when the maximal flight time budget B is depleted, where flight times are calculated using Eq. (4.2). At each time step t, the tree search is executed to plan the next UAV position $\mathbf{p}_t$ based on the current belief state s_t in Eq. (4.6), where we use the actor-critic network to guide the search. The tree search returns a sampling policy $\boldsymbol{\pi}(s_t)$ the next measurement position $\mathbf{p}_t$

is drawn from. Additionally, the executed path's information value $I(\psi_{\pi,t})$ is computed based on the received immediate rewards. Belief states s_t, sampling policies $\pi(s_t)$, and information values $I(\psi_{\pi,t})$ are stored in an experience buffer.

In the following Sec. 4.2.3, we detail the actor-critic network-based tree search used to derive the sampling policy. Subsequently, we describe our actor-critic network architecture and explain the training procedure in Sec. 4.2.4.

4.2.3 Sampling-Based Planning with Neural Networks

At each time step t of a terrain monitoring mission, we execute a tree search formalised in Alg. 4 similar to MCTS [13] that plans the next measurement position $\mathbf{p}_t$ based on the belief state s_t in Eq. (4.6). The search is guided by an actor-critic neural network parameterised by $\theta \in \mathbb{R}^D$ that processes the belief state s_t and outputs prior sampling probabilities $\boldsymbol{\pi}_\theta(s_t)$ over all possible next measurement positions $\mathbf{p}_t \in \mathcal{P}_t$ and an estimated information value $V_\theta(s_t) \approx I(\psi_{\pi,t})$.

The tree search is executed iteratively for a fixed number of iterations $S \in \mathbb{N}$ from a root node n_0 representing the current belief state s_t in Line 1. A node i at tree depth d is denoted as $n_{d,i}$ and consists of its respective belief state s_{t+d}, subsequently reachable child nodes $n_{d+1,j} \in \mathcal{N}_{d+1}^i$ at depth $d+1$, and total value and visit counts initialised as $V(n_{d,i}) = 0$ and $N(n_{d,i}) = 0$. In each iteration, in Line 22, the tree is traversed until the UAV's flight budget is exceeded or the maximum path length L, we plan ahead is reached. If a leaf node $n_{d,i}$ at depth $d < L$ with no child nodes $\mathcal{N}_{d+1}^i = \emptyset$ is reached in Line 23, we expand the search tree by adding child nodes $n_{d+1,k}$ for each possible next measurement position $\mathbf{p}_{t+d}$ reachable within the remaining UAV flight time B_t. Then, we estimate the information value $I(\psi_{\pi,t}) \approx V_\theta(s_{t+d})$ of leaf node $n_{d,i}$ using the critic network, see Line 28. In this way, we avoid the costly simulation of future measurements and map updates up to the planning horizon $t + L$ required to evaluate the leaf node's information value in non-learning-based search methods.

If node $n_{d,i}$ is not a leaf, i.e. $\mathcal{N}_{d+1}^i \neq \emptyset$, we select a child node $n_{d+1,k} \in \mathcal{N}_{d+1}^i$ to traverse the tree along a subsequent measurement position in Line 29 that maximises the probabilistic upper confidence tree (PUCT) bound [144]

$$\mathrm{PUCT}(n_{d+1,k}) = Q(s_{t+d}, \mathbf{p}_{t+d}) + \boldsymbol{\pi}_\theta(s_{t+d})_k \, U(n_{d+1,k}), \tag{4.14}$$

$$Q(s_{t+d}, \mathbf{p}_{t+d}) = R(s_{t+d}, \mathbf{p}_{t+d}, s_{t+d+1}) + \gamma \, \frac{V(n_{d+1,k})}{N(n_{d+1,k})}, \tag{4.15}$$

$$U(n_{d+1,k}) = \frac{\sqrt{N(n_{d,k})}}{1 + N(n_{d+1,k})} \left(c_1 + \log\left(\frac{N(n_{d,k}) + c_2 + 1}{c_2} \right) \right), \tag{4.16}$$

where $Q(s_{t+d}, \mathbf{p}_{t+d})$ is the information value of starting at node $n_{d,i}$ and choosing the next measurement position $\mathbf{p}_{t+d}$ leading to child node $n_{d+1,k}$. The child

Algorithm 4 Actor-Critic Network-based Tree Search

procedure PLAN(s_t, S)
  $s \leftarrow 1$ ▷ counts search iterations
  Initialise search root node n_0 with state s_t
  Initialise total node value $V(n_0) = 0$ and visit counts $N(n_0) = 0$
  Initialise empty set of children $\mathcal{N}_1^0 = \emptyset$
  while $s \leq S$ **do** ▷ performs S search iterations
    SEARCH($n_0, 0$) ▷ traverses and updates tree from root to leaf node
    $s \leftarrow s + 1$
  return $N(n_{1,k})\ \forall n_{1,k} \in \mathcal{N}_1^0$ ▷ visist counts of root node's children

procedure SIMULATE($s_t, \mathbf{p}$)
  Get terrain points $\mathbf{x}_i \in \xi$ for all pixels i from measurement position $\mathbf{p}$
  $\varepsilon_i \sim \mathcal{N}(0, \nu(\mathbf{p}))$ for all pixels i ▷ sample i.i.d. altitude-dependent noise
  $\mathbf{z}_{i,t+1} \leftarrow \boldsymbol{\mu}_t(\mathbf{x}_i) + \varepsilon_i$ for all pixels i ▷ simulate measurements from map
  Update Kalman filter to $\mathcal{N}(\boldsymbol{\mu}_{t+1}, \mathbf{P}_{t+1})$ based on $\mathcal{N}(\boldsymbol{\mu}_t, \mathbf{P}_t)$ and $\mathbf{z}_{t+1}$
  Construct belief state s_{t+1} as in Eq. (4.6) with $B_{t+1} = B_t - c(\mathbf{p}_{t-1}, \mathbf{p})$
  Compute reward $R(s_t, \mathbf{p}, s_{t+1})$ using Eq. (4.12)
  return belief state s_{t+1} and reward $R(s_t, \mathbf{p}, s_{t+1})$

procedure SEARCH($n_{d,i}, d$)
  Extract state s_{t+d} from node $n_{d,i}$ ▷ i-th node at tree depth d
  Extract children $\mathcal{N}_{d+1}^i$ of $n_{d,i}$ ▷ nodes at depth $d+1$ reachable from $n_{d,i}$
  if $B_t \leq 0$ or $d = L$ **then return** 0
  if $n_{d,i}$ is leaf node **then** ▷ check for $\mathcal{N}_{d+1}^i = \emptyset$
    for $\mathbf{p}_{t+d} \in \mathcal{P}_{t+d}$ reachable within remaining budget B_t **do**
      Add new child node $n_{d+1,k}$ with position $\mathbf{p}_{t+d}$
    $V(n_{d,i}) \leftarrow V_\theta(s_{t,d})$ ▷ Infer information value from critic network
    $N(n_{d,i}) \leftarrow 1$
    return $V_\theta(s_{t,d})$
  $n_{d+1,j} \leftarrow \text{argmax}_{n_{d+1,k} \in \mathcal{N}_{d+1}^i} \text{PUCT}(n_{d+1,k})$ ▷ using Eq. (4.14)
  $s_{t+d+1}, R(s_{t+d}, \mathbf{p}_{t+d}, s_{t+d+1}) \leftarrow$ SIMULATE($s_{t+d}, \mathbf{p}_{t+d}$)
  $V(n_{d,i}) \leftarrow V(n_{d,i}) + R(s_{t+d}, \mathbf{p}_{t,d}, s_{t+d+1}) + \gamma$SEARCH($n_{d+1,j}, d+1$)
  $N(n_{d,i}) \leftarrow N(n_{d,i}) + 1$

node's information value is estimated based on its total value $V(n_{d+1,k})$ normalised over the number of search iterations $N(n_{d+1,k})$ traversing through $n_{d+1,k}$. The constants $c_1, c_2 \in \mathbb{R}^+$ are factors weighing the exploration term $U(n_{d+1,k})$ balancing between choosing measurement positions with high information val-

ues $Q(s_{t+d}, \mathbf{p}_{t+d})$ and measurement positions that are traversed less frequently. In contrast to non-learning-based search methods, we do not uniformly explore measurement positions based on their visit counts but favour child nodes $n_{d+1,k}$ with high prior sampling probability $\boldsymbol{\pi}_\theta(s_{t+d})_k$ predicted by our actor network. In this way, we improve the tree search's sample efficiency as it is guided towards paths with high estimated information value. Similarly to Silver et al. [152], we add Dirichlet-noise to the root node's sampling probabilities $\boldsymbol{\pi}_\theta(s_t)$ to foster exploration of immediate next measurement positions during training.

For a chosen child node $n_{d+1,j}$ with position $\mathbf{p}_{t+d}$ maximising the PUCT in Eq. (4.14), we estimate the immediate reward $R(s_{t+d}, \mathbf{p}_{t+d}, s_{t+d+1})$ by simulating a measurement at $\mathbf{p}_{t+d}$ in Line 13. As the UAV cannot collect real measurements during planning, we query the Kalman filter's mean $\boldsymbol{\mu}_{t+d}(\mathbf{x}_i)$ at measurement points $\mathbf{x}_i$ projected from $\mathbf{p}_{t+d}$ on the terrain to get the maximum a posteriori estimate of a potential future measurement $\mathbf{z}_{t+d}$ at $\mathbf{p}_{t+d}$. Additionally, we sample zero-centered altitude-dependent measurement noise ε from $\mathcal{N}(0, \nu(\mathbf{p}_{t,d}))$ considering the camera's noise. We update the map based on the simulated measurement $\mathbf{z}_{t+d}$ using the Kalman filter equations described in Sec. 4.1.1. Based on the updated map belief, we construct the state s_{t+d+1} of child node $n_{d+1,j}$ as in Eq. (4.6) and compute the reward $R(s_{t+d}, \mathbf{p}_{t+d}, s_{t+d+1})$ using Eq. (4.12). We compute the path's information value from child node $n_{d+1,j}$ on, which is used to derive the path's information value starting in parent node $n_{d,i}$, recursively repeating the traversal in Line 31 until a leaf node is reached.

After S search iterations, we derive the sampling policy $\boldsymbol{\pi}(s_t)$ of the root node's state s_t for each child node $n_{1,k} \in \mathcal{N}_1^0$ based on its visit counts as

$$\boldsymbol{\pi}(s_t)_k = \frac{N(n_{1,k})^{1/\tau}}{\sum_{n_{1,l} \in \mathcal{N}_1^0} N(n_{1,l})^{1/\tau}}, \tag{4.17}$$

where $\mathcal{N}_1^0$ are the root's child nodes with associated measurement positions. Sampling policies are uniform for hyper-parameter $\tau \to \infty$, and policies peak to $\operatorname{argmax}_k \boldsymbol{\pi}(s_t)_k$ as $\tau \to 0$. During training, $\mathbf{p}_t$ is sampled from $\boldsymbol{\pi}(s_t)$ with $\tau > 0$, resulting in next state s_{t+1} and reward $R(s_t, \mathbf{p}_t, s_{t+1})$. State s_t, sampling policy $\boldsymbol{\pi}(s_t)$, and reward $R(s_t, \mathbf{p}_t, s_{t+1})$ are stored in an experience buffer $\mathcal{D}$. During deployment, we choose the most informative next measurement position $\mathbf{p}_t$ setting $\tau = 0$ and not adding Dirichlet-noise to $\boldsymbol{\pi}_\theta(s_t)$. In the following Sec. 4.2.4, we detail the actor-critic network training using gathered experience $\mathcal{D}$.

4.2.4 Network Architecture and Training

Our actor-critic network is parameterized by $\theta \subset \mathbb{R}^D$ with its architecture being depicted in Fig. 4.3. It predicts prior sampling probabilities $\boldsymbol{\pi}_\theta(s_t)$ and information value $V_\theta(s_t) \approx I(\psi_{\pi,t})$ of a path starting in state s_t following the tree

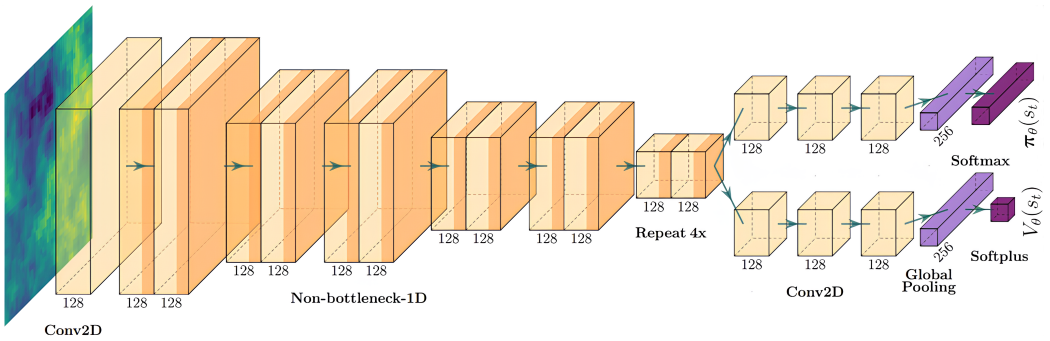


Figure 4.3: Our actor-critic convolutional neural network processes belief states s_t. We leverage an ERFNet encoder [131] with 10 residual blocks providing shared representations for predicting prior sampling probabilities $\boldsymbol{\pi}_\theta(s_t)$ and information value $V_\theta(s_t)$. Both heads comprise three convolutional blocks and global pooling to make the convolutional neural network map dimension-agnostic. Finally, fully connected layers project to predictions $\boldsymbol{\pi}_\theta(s_t)$ and $V_\theta(s_t)$.

search-induced sampling policy π in Eq. (4.17). We design a convolutional neural network that processes a belief state s_t defined in Eq. (4.6) with a shared encoder. We leverage non-bottleneck-1D blocks proposed by Romera et al. [131] to reduce inference time. The encoder is followed by two separate prediction heads for prior sampling probabilities $\boldsymbol{\pi}_\theta(s_t)$ and information value $V_\theta(s_t)$. Both heads consist of three blocks with 2D convolution, batch norm, and SiLU activations. The last block's feature maps in each head are flattened to fixed-dimensional latent vectors using global average and max pooling before applying final fully connected layers to ensure the architecture is map dimension-agnostic.

The network's input consists of (i) the covariance matrix $\mathbf{P}_t$, (ii) the remaining budget B_t, (iii) the UAV position $\mathbf{p}_t$, (iv) a cost grid map with $c(\mathbf{p}_t, \mathbf{x}_i)$ approximating the UAV's flight time from position $\mathbf{p}_t$ to a measurement position above $\mathbf{x}_i \in \mathcal{X}$, and (v) $\hat{\mathcal{X}}_{I,t}$ indicating areas that are likely of interest according to Eq. (4.3). Additionally, we input the previous two covariance matrices $\mathbf{P}_{t-2}, \mathbf{P}_{t-1}$, measurement positions $\mathbf{p}_{t-2}, \mathbf{p}_{t-1}$, and remaining budgets B_{t-2}, B_{t-1}. All inputs are min-max normalised and expanded to the covariance matrix dimensions.

The convolutional neural network is trained with stochastic gradient descent using a one-cycle learning rate over three epochs [156]. We sample states s_t, sampling policies $\boldsymbol{\pi}(s_t)$, and received rewards $R(s_t, \mathbf{p}_t, s_{t+1})$ from experience buffer $\mathcal{D}$. The sampling policies are prediction targets for the prior sampling probabilities $\boldsymbol{\pi}_\theta(s_t)$. Furthermore, we compute L-step information values $V_\pi(s_t) = \sum_{d=0}^{L-1} \gamma^d R(s_{t+d}, \mathbf{p}_{t+d}, s_{t+d+1})$ based on the L rewards subsequently received from state s_t on, where L is the planning horizon of the tree search, acting as prediction targets for estimated information values $V_\theta(s_t)$. The actor-critic network is trained to minimise the loss function

$$\mathcal{L}(\theta) = \mathop{\mathbb{E}}_{\substack{s_t, \boldsymbol{\pi}(s_t), \\ V_\pi(s_t) \sim \mathcal{D}}} \left[\alpha (V_\theta(s_t) - V_\pi(s_t))^2 - \beta \boldsymbol{\pi}(s_t)^T \log\left(\boldsymbol{\pi}_\theta(s_t)\right) \right] + \lambda \|\theta\|^2, \quad (4.18)$$

where α is the critic loss coefficient, β is the actor loss coefficient, and $\lambda \geq 0$ is the weight decay coefficient. After re-training is completed, we again generate new missions as explained in Sec. 4.2.2 using the re-trained actor-critic network θ in the tree search to plan next measurement positions as detailed in Sec. 4.2.3.

4.2.5 Implementation Details

In the following, we describe the implementation details of our tree search algorithm introduced in Sec. 4.2.3 and of our actor-critic network architecture described in Sec. 4.2.4 that aim to improve information gathering performance.

A major shortcoming of neural network-guided tree search [153] is that the sampling policy in Eq. (4.17) reflects the search exploration dynamics induced by the PUCT in Eq. (4.14). The root's child node visit counts $N(n^0_{1,k})$ do not necessarily capture the information values $Q(s_t, \mathbf{p}_t)$ for possible next measurement positions $\mathbf{p}_t$ from the root node's state s_t given a finite number of search iterations. Hence, the tree search tends to overemphasize initially explored measurement positions $\mathbf{p}_t$, leading to bias in collected experience $\mathcal{D}$ and in the trained actor-critic network. Next, we introduce techniques to counteract these problems.

To avoid overemphasizing initially explored measurement positions, exploration while traversing the search tree is desirable. However, increasingly exploiting known informative positions is beneficial in later re-training iterations. Thus, we exponentially decay the exploration constant c_1 in Eq. (4.14) after each re-training. Similarly, we introduce an exponentially decaying amount of Dirichlet noise added to the root node's prior sampling probabilities $\boldsymbol{\pi}_\theta(s_t)$ to gradually exploit the learned prior sampling probabilities.

Additionally, we introduce an experience buffer $\mathcal{D}$ of increasing size $|\mathcal{D}|$ to accelerate training [96]. On the one hand, a substantial amount of experience is required to train the convolutional neural network on a diverse set of simulated information-gathering missions. On the other hand, in earlier training stages, a small experience buffer accelerates network improvements while larger buffers in later training stages ensure data diversity. Hence, we linearly increase the experience buffer size in each iteration of generating new episodes.

Moreover, we adapt two techniques introduced by Wu [180] to improve the tree search's sample efficiency and the network's architecture. First, forced playouts and policy pruning decouple the tree search's exploration dynamics and sampling policies $\boldsymbol{\pi}(s_t)$. While traversing the search tree, under-explored child nodes $n^0_{1,k}$ of the root node n_0 are chosen by setting $\text{PUCT}(n^0_{1,k}) = \infty$ in Eq. (4.14). In Eq. (4.17), these child node visits $N(n^0_{1,k})$ are subtracted again unless measurement position $\mathbf{p}_{t+1}$ of child node $n^0_{1,k}$ led to a high information value. Second, we use multiple global pooling bias blocks in the actor-critic network's shared encoder. These blocks split the feature maps of previous non-bottleneck-1D blocks

into two disjoint subsets. Global adaptive pooling is applied to the first subset of feature maps, and a fully connected layer projects the pooled features to a latent vector with dimensions equal to the number of feature maps in the second subset. The projected latent vector is then used as a channel-wise bias for the second subset of feature maps. This enables our convolutional neural network to focus more easily on spatially local and global features required for adaptive IPP. For details on forced layouts and global pooling bias blocks, we refer to Wu [180].

4.3 Experimental Evaluation

The experiments are designed to evaluate our approach and experimentally investigate the claims made in this chapter. First, we show that our RL-based adaptive IPP method accelerates the replanning of paths compared to non-learning-based state-of-the-art methods, resulting in higher information-gathering performance in simulated UAV-based terrain monitoring missions as shown in Sec. 4.3.2. Second, in Sec. 4.3.3, we validate that our method outperforms traditionally used pre-computed coverage paths in a previously unseen real-world dataset-based field surface temperature mapping mission. Third, we verify the individual components of our adaptive IPP algorithm in an ablation study discussed in Sec. 4.3.4.

4.3.1 Experimental Setup

Mission setup. The procedure for simulating UAV-based terrain monitoring missions is detailed in Sec. 4.2.2. Our simulation setup considers terrains $\xi \subset \mathbb{R}^2$ with 2D scalar feature fields $F : \xi \to [0, 1]$, where scalar values $F(\mathbf{x})$ are assumed to be normalised between 0 and 1. We discretise the terrain ξ into a 2D grid map $\mathcal{X}$ to perform Kalman filter map updates $\hat{F}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{P}_t)$ as described in Sec. 4.1.1. We set the interesting feature threshold $f_{th} = 0.4$ and randomly split the terrain ξ into high- and low-value regions to create areas of interest ξ_I following Popović et al. [126]. The UAV's measurement positions are defined by a discrete 2.5D lattice above the terrain ξ [126]. The lattice mirrors the 15×15 grid map $\mathcal{X}$ on two altitude levels at 8 m and 14 m. We use the altitude-dependent inverse sensor model by Popović et al. [126] to simulate camera noise, assuming a downwards-facing square camera footprint with 60° field of view. The terrain map's Gaussian process prior is defined by a constant mean function with $m(\mathbf{x}) = 0.5$ for all $\mathbf{x} \in \xi$. The Gaussian process uses the Matérn 3/2 kernel with length scale 3.67, signal variance 1.82, and noise variance 1.42 by maximizing log marginal likelihood over independent maps [126]. The UAV's mission budget is $B = 150\,\text{s}$ flight time, its initial position is $\mathbf{p}_0 = (2, 2, 14)\,\text{m}$, and its constant acceleration-deceleration is $\pm u_a = 2\,\text{m/s}^2$ with maximum speed $u_v = 2\,\text{m/s}$.

Hardware. We execute the UAV-based terrain monitoring missions on a laptop with a 1.8 GHz Intel i7 CPU, 16 GB memory without GPU acceleration to avoid unfair advantages in inference speed of our actor-critic network, trained offline on a single machine with a 2.2 GHz AMD Ryzen 9 3900X with 64 GB memory and a NVIDIA GeForce RTX 2080 Ti GPU.

Evaluation metrics. We repeat 10 UAV-based terrain monitoring missions and report means and standard deviations. All performance metrics are computed over areas of interest $\xi_I \subset \xi$ to assess the methods' adaptive replanning capabilities. The adaptive IPP performance of a method is assessed by the map uncertainty and error in areas of interest as in prior works [30, 61, 126]. At time step t, the current map belief's $\hat{F}_t \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{P}_t)$ uncertainty is defined as

$$\mathrm{Unc}(\hat{F}_t, \xi_I) = \sum_{\mathbf{x}_i \in \mathcal{X}} \mathbb{I}_{\mathbf{x}_i \in \xi_I} \mathbf{P}_{t,i,i} \,, \tag{4.19}$$

where $\mathbb{I}_{\mathbf{x}_i \in \xi_I}$ is one, if grid cell $\mathbf{x}_i \in \mathcal{X}$ is part of an area of interest ξ_I, else zero, and $\mathbf{P}_{t,i,i}$ is the grid cell's $\mathbf{x}_i$ current map variance. In this way, we compute the trace of the covariance matrix of the current map belief in areas of interest ξ_I. Lower map uncertainty values indicate better adaptive IPP performance.

The map error at time step t is captured by computing the root mean squared error (RMSE) between the map belief $\hat{F}_t \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{P}_t)$ and the ground truth scalar feature field F in areas of interest ξ_I as

$$\mathrm{RMSE}(\hat{F}_t, F, \xi_I) = \sqrt{\frac{1}{N_I} \sum_{\mathbf{x}_i \in \mathcal{X}} \mathbb{I}_{\mathbf{x}_i \in \xi_I} (\boldsymbol{\mu}_t(\mathbf{x}_i) - F(\mathbf{x}_i))^2} \,, \tag{4.20}$$

where N_I is the number of grid cells in $\mathcal{X}$ belonging to areas of interest ξ_I, $\boldsymbol{\mu}(\mathbf{x}_i)$ is the map's current mean estimate for a feature value at grid cell $\mathbf{x}_i$ and $F(\mathbf{x}_i))$ is the ground truth feature value of the grid cell. Lower map errors indicate better adaptive IPP performance. We evaluate map uncertainty and error over the spent mission time as in prior works [30, 61, 126]. The spent mission time incorporates the path travel time and planning runtime to evaluate performance under limited onboard resources. A faster reduction of map uncertainty and error over the spent mission time indicates better adaptive IPP performance.

Baselines. Our RL-based adaptive IPP approach is compared against state-of-the-art non-learning-based adaptive IPP methods. All baselines use the same reward function defined in Eq. (4.12) and simulate future measurements and map updates during planning using the current map belief as described in Sec. 4.2.3. We implement (i) MCTS, a rollout-based solver to plan the next measurement position based on sampled future finite-horizon paths using progressive widening of the action space [160] and a generalized cost-benefit rollout policy for adaptive IPP scenarios as proposed by Choudhury et al. [30]; (ii) CMA-ES fine-tunes an initially greedily chosen path over the 2.5D lattice in the continuous

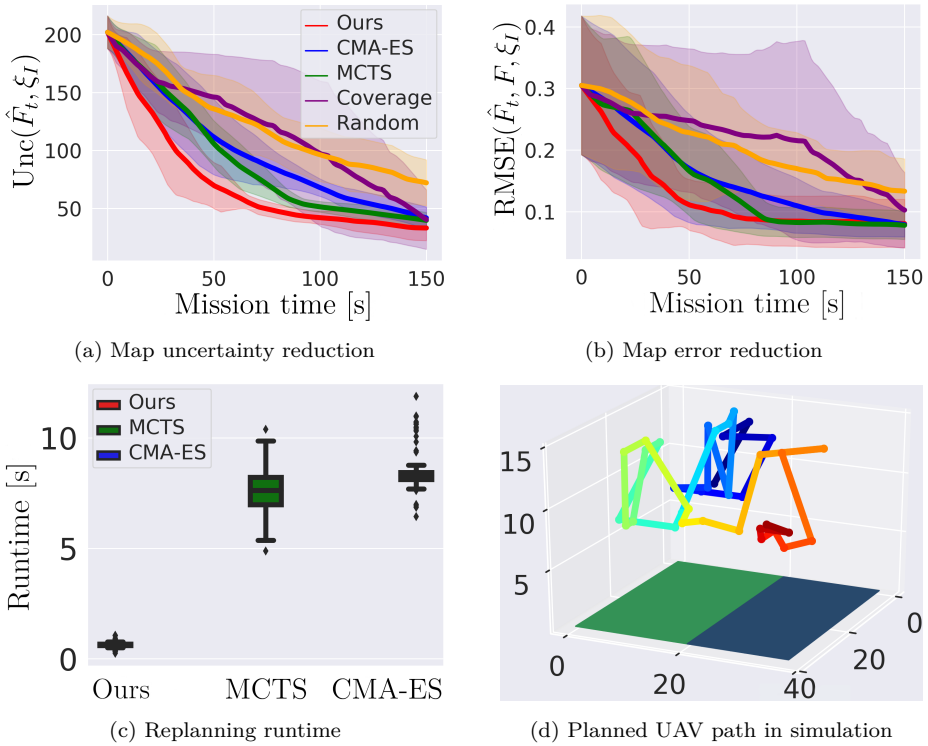
3D workspace using covariance matrix adaptation evolution strategy (CMA-ES) as proposed by Popović et al. [126]. Both planners consider a 5-step planning horizon and we set CMA-ES hyper-parameters to 45 iterations, 12 offsprings, and $(4, 4, 3)$ m coordinate-wise step size in line with Popović et al. [126]. Furthermore, we investigate two non-adaptive baseline planners that pre-compute paths before deployment, thus not inducing any replanning runtime costs during deployment. The random strategy chooses the next measurement position at random while the coverage strategy pre-computes equidistantly spaced measurement position at a fixed 8 m altitude to spatially cover the entire terrain.

4.3.2 Simulation Results

The first set of experiments evaluates our RL-based adaptive IPP approach against state-of-the-art non-learning-based methods in simulated terrain monitoring scenarios to investigate our first claim. In line with Popović et al. [126], we randomly split the terrain into low- and high-value areas of interest as illustrated in Fig. 4.4d to assess the methods' adaptive replanning capabilities. Our results show that our RL-based tree search approach accelerates replanning, resulting in higher information-gathering performance in the simulated UAV-based terrain monitoring missions compared to state-of-the-art baseline methods.

Fig. 4.4 illustrates the information-gathering performance of our method (red) compared to state-of-the-art adaptive IPP methods. As in prior works [30, 61, 102, 126], all methods adaptively replanning paths based on the current map belief show on average faster reduction of the map uncertainty and error in areas of interest. Although some coverage paths quickly gather information about areas of interest, we observe high variance in the coverage strategy's information-gathering performance as it cannot adaptively focus on areas of interest. This validates that adaptive IPP methods are required to achieve higher information-gathering performance than traditionally used pre-computed paths.

Particularly, our method reduces the map uncertainty and error in areas of interest on average faster than the state-of-the-art adaptive CMA-ES (blue) and MCTS (green) replanning strategies. Additionally, our method substantially reduces replanning runtime, achieving a speedup factor of $8 - 10\times$ compared to non-learning-based CMA-ES and MCTS planning methods. These results highlight the improved efficiency in our tree search and confirm that the actor-critic network can learn informative measurement positions from training in diverse simulated terrain monitoring missions. Fig. 4.4d shows a simulated terrain monitoring mission split into low (blue) and high-value areas of interest (green) and the path planned by our method. The UAV uses most of its mission budget for collecting measurements of areas of interest as they are discovered. This qualitatively validates the learned adaptive planning strategy of our method.



(a) Map uncertainty reduction

(b) Map error reduction

(c) Replanning runtime

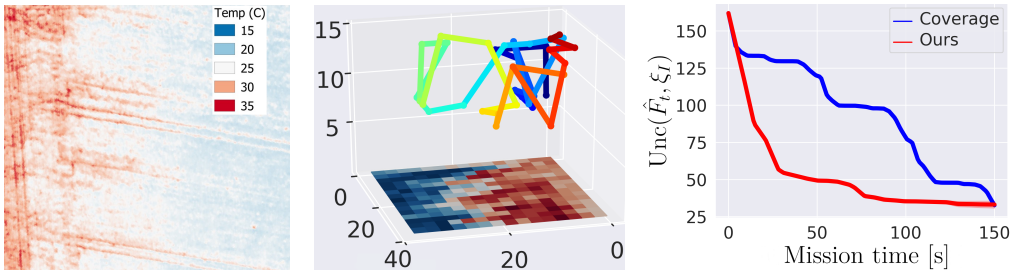
(d) Planned UAV path in simulation

Figure 4.4: Our approach against state-of-the-art adaptive IPP methods in simulated UAV-based terrain monitoring missions. Solid lines indicate means over 10 missions, and shaded regions indicate the standard deviations. (a+b) On average, our RL-based approach ensures the fastest map uncertainty and error reduction in areas of interest over the mission time. (c) Furthermore, replanning runtime is reduced by a factor of $8 - 10\times$ compared to adaptive baselines. (d) The planned path (evolving over time from blue to red) validates the adaptive behaviour of our approach, exploring the terrain while focusing on high-value areas (green).

4.3.3 Results on Real-World Surface Temperature Data

The second set of experiments evaluates our method in a UAV-based terrain monitoring scenario using a real-world orthomosaic dataset to assess our second claim. Our method outperforms traditionally used pre-computed coverage paths in terrain monitoring missions on previously unseen real-world terrain datasets.

We demonstrate our approach using real-world surface temperature data of a crop field. The data was collected in a 40×40 m large area of a crop field nearby Forschungszentrum Jülich, Germany (50.87° lat., 6.44° lon.) on July 20, 2021. The data was acquired with a DJI Matrice 600 UAV carrying a Vue Pro R 640 thermal sensor. The collected images were then processed using Pix4D software to generate an orthomosaic representing the surface temperature in our simulation as illustrated in Fig. 4.5a. We consider areas with surface temperatures above 25°C to be of interest as hot field regions could correlate with drought stress of crops that might require intervention. We compare our approach against a non-adaptive coverage path traditionally used to monitor arable fields [48].



(a) Surface temperature data (b) Planned UAV path and map (c) Map uncertainty reduction

Figure 4.5: Real-world dataset-based UAV surface temperature mapping scenario. (Left) The surface temperature orthomosaic of a crop field with high temperature (red) indicating areas of interest. (Centre) The mapped surface temperature and path planned using our approach (evolving over time from blue to red). (Right) Our RL-based approach ensures fast map uncertainty reduction in high-temperature areas, outperforming traditionally used coverage paths.

Fig. 4.5b shows the planned path above the crop field and the resulting terrain map using our method. Our method explores the terrain and allocates more of the mission budget for collecting measurements in areas of interest with high surface temperatures as they are discovered (red). This qualitatively verifies the adaptive replanning behaviour of our learned tree search on previously unseen real-world terrain data. Furthermore, Fig. 4.5c quantitatively shows that our approach ensures fast map uncertainty reduction in areas of high surface temperature. The coverage path cannot adapt its behaviour based on already collected measurements and thus fails to quickly reduce map uncertainty in areas of high surface temperature. These results verify the successful transfer of our actor-critic network trained in simulation to a real-world dataset and demonstrate the tree search's benefits over traditionally used pre-computed monitoring approaches.

4.3.4 Ablation Study

The third set of experiments is an ablation study designed to evaluate the tree search and actor-critic network components detailed in Sec. 4.2.5, investigating our third claim. Our results verify that the individual components often improve the map uncertainty and error reduction, validating our algorithm design.

We perform an ablation study comparing our approach to versions of itself (i) removing proposed tree search components detailed in Sec. 4.2.5 and (ii) changing the actor-critic architecture introduced in Sec. 4.2.4. We reduce the grid map's $\mathcal{X}$ resolution from 15×15 cells used for experiments in Sec. 4.3.2 and Sec. 4.3.3 to 10×10 cells to retrain the different variants more efficiently. We compare them in simulated terrain monitoring missions as in Sec. 4.3.2.

The ablation study results are summarised in Tab. 4.1. Considering not only the current state information as input to the actor-critic network but also provid-

Variant	Unc. ↓ 33%	67%	100%	RMSE ↓ 33%	67%	100%	Time ↓ [s]
Our approach in Sec. 4.2	**73.61**	31.83	**12.44**	**0.15**	0.09	**0.05**	0.64
(i) w/ constant buffer size	83.25	50.17	24.68	0.16	0.12	0.09	0.68
(i) w/ fixed exploration constants	95.27	39.46	21.86	0.20	0.11	0.08	0.65
(i) w/o forced playouts	79.23	**28.53**	22.62	0.18	0.09	0.07	0.66
(ii) w/o glob. pool. bias blocks	103.58	45.78	31.44	0.19	0.11	0.10	0.64
(ii) 5 blocks in encoder	82.90	29.94	17.94	0.16	**0.08**	0.07	**0.55**
(ii) w/o input feature history	102.40	40.48	31.33	0.20	0.10	0.09	0.66

Table 4.1: Ablation study results of our approach presented in Sec. 4.2. We systematically (i) remove algorithm components and (ii) change the actor-critic architecture to quantify their impact on the adaptive IPP performance. The map uncertainty (Unc.) and error (RMSE) are evaluated after 33%, 67%, and 100% spent mission time. Faster map uncertainty and error reduction over time indicate stronger performance. Our approach introduced in Sec. 4.2 achieves the fastest and most stable reductions in uncertainty and RMSE over the mission time.

ing it with a history of the previous two map beliefs and UAV positions results in facilitated learning and, thus, substantially faster map uncertainty and error reduction. Similarly, using global pooling bias blocks in the actor-critics encoder proposed by Wu [180] enhances learning, reflected in noticeably improved information-gathering efficiency. Reducing the actor-critic network's size to 5 instead of 10 encoder blocks has a less pronounced effect on the planning performance but still leads to higher remaining map uncertainty and error after termination of the mission. Moreover, the algorithm's components influencing the tree search behaviour, i.e. using dynamically adjusted exploration constants while traversing the search and using forced playouts of the root node's under-explored child nodes, both lead to lower final map uncertainty and error in areas of interest. Finally, dynamically increasing the size of the experience buffer $\mathcal{D}$ while generating training episodes results in faster map uncertainty and error reduction as well. Overall, these results show that our algorithm and actor-critic network design choices consistently improve the information-gathering efficiency in simulated UAV-based terrain monitoring missions.

4.4 Conclusion

To gather information in unknown environments, robots must decide onboard where to move next to collect new measurements that improve their understanding of the environment. Recent adaptive IPP methods enable robots to plan their next measurement positions based on already gathered measurements. However, these approaches involve compute-expensive procedures to replan paths, resulting in decreased information-gathering efficiency on resource-constrained robots.

To address this issue, we proposed a new adaptive IPP approach that combines sampling-based planning with RL to accelerate onboard replanning of paths on resource-constrained UAVs. To achieve this, we formalise the connection between the adaptive IPP problem for information gathering in terrain monitoring and the general RL problem. We then use this established connection to offline-train a neural network in simulated UAV-based terrain monitoring missions, which steers the sampling of potential future candidate paths and estimates their expected information value based on the robot's current understanding of the environment. Our experimental results show that our RL-based adaptive IPP method accelerates the replanning of paths, resulting in higher information-gathering performance than state-of-the-art non-learning-based adaptive methods in simulated UAV-based terrain monitoring missions. Furthermore, our method outperforms traditionally used pre-planned coverage strategies on previously unseen real-world crop field surface temperature data-based mapping missions.

These results suggest that, in response to the first research question posed in this thesis, our approach combining RL and sampling-based planning presents one possible way forward to increase the compute efficiency of adaptive IPP algorithms without sacrificing information-gathering performance. However, our method is trained for specific terrain monitoring missions, restricting its application to monitoring continuous-valued terrain features, such as surface temperature. Additionally, we trained our planning algorithm on static user-defined mission characteristics, such as fixed interesting feature value thresholds qualifying an area to be of interest for precise mapping. Hence, changing user-defined mission characteristics might require re-training. Moreover, other information-gathering missions, such as monitoring streets, vegetation, and buildings in a city for urban planning purposes, require monitoring discrete-valued semantic information and thus require re-designing the method. In Chap. 5, we address these limitations by proposing a novel adaptive IPP formulation that unifies this broad family of information-gathering missions. Similar to this chapter, we then train a single adaptive IPP policy with RL applicable to a large variety of terrain monitoring missions using our unified formulation.

Chapter 5

Map-Agnostic Policies for Adaptive Informative Path Planning

Recent advances have prompted a series of non-learning-based methods that address the adaptive IPP problem [61, 63, 102, 117, 117, 126]. To adaptively replan paths, non-learning-based adaptive IPP methods rely on evaluating the potential information value of many possible future paths in the environment. As computing the information value involves costly simulation of expected future measurements and map updates, these procedures are computationally expensive. This makes it challenging to deploy non-learning-based methods on resource- and compute-constrained robots while ensuring frequent replanning of paths. To overcome these issues, learning-based adaptive IPP approaches, such as ours described in Chap. 4 have been proposed. These methods train planning policies offline in simulation and perform compute-efficient policy inference at deployment [18, 19, 27, 98, 116, 135, 174].

The majority of approaches consider different to-be-mapped information, i.e. environmental features, of interest during a mission. Mapping continuous-valued features, e.g. bacteria levels [61] or signal strength [102], is commonly performed using Gaussian processes [19, 27, 61, 102, 119, 171, 174] or Kalman filters [126, 135] as map representations. Mapping discrete-valued features, e.g. semantically segmenting crops and weeds in arable fields [175] or analysing rural area land use [126], is commonly performed using grid maps [18, 98, 116, 126, 175]. These approaches require re-designing the environment representation used for planning as the to-be-mapped environmental features, and thus, the map representations change. Furthermore, learning-based approaches, including the approach discussed in Chap. 4, are not only specifically designed for but also trained on a single environment map representation. This prohibits their direct application

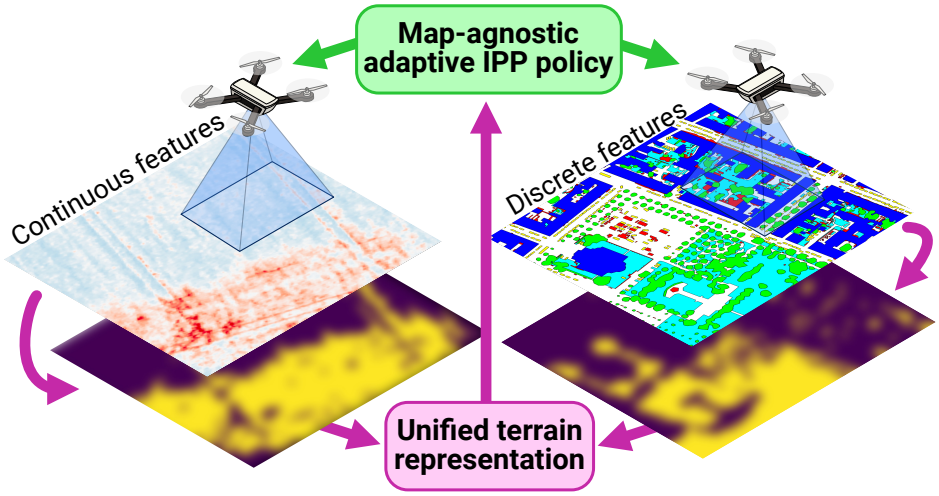


Figure 5.1: Robots perform continuous- or discrete-valued terrain feature monitoring missions, e.g. mapping surface temperature or urban semantics. We transform mission-specific terrain map representations, e.g. Gaussian processes or occupancy grid maps, into a novel unified state representation for adaptive informative path planning (IPP). In this way, we design and train a single map-agnostic planning policy applicable to largely varying terrain monitoring missions.

to a larger variety of robotic information-gathering missions since these methods require re-training planning policies as map representations change.

To address this issue, we argue in this chapter that the broad pool of existing adaptive IPP approaches should be viewed along two dimensions: the map-specific formulation modelling the adaptive IPP problem and the algorithm used to offline-train or online-search the planning policy. The formulation of the adaptive IPP problem is the most critical design decision to ensure the unified applicability of planning policies across various monitoring missions. This motivates the need for a map-agnostic formulation of the adaptive IPP monitoring problem that directly integrates with any (non)-learning-based policy search algorithm used for adaptive IPP. Particularly, this formulation ensures training and deploying learned policies in largely varying monitoring missions.

The main contribution of this chapter is such a novel map-agnostic formulation of the adaptive IPP problem for terrain monitoring. Our formulation unifies continuous-valued, i.e. regression, and discrete-valued, i.e. classification, terrain feature monitoring for adaptive IPP policies as illustrated in Fig. 5.1. To achieve this, we unify the planning algorithm's state space representation across terrain map representations used for replanning paths online. Based on this unified planning state space and a new reward function for adaptive IPP, we train and deploy a generally applicable planning policy on previously unmet variations of terrain monitoring missions using RL. To this end, we condition the learned planning policy on the user-defined and map-specific mission hyperparameters to increase performance across diverse monitoring missions.

Overall, we make the following claims in this chapter. First, our map-agnostic planning policy trained and deployed on vastly varying simulated terrain monitoring missions performs on par or better than state-of-the-art map-specifically trained policies and non-learning-based adaptive IPP approaches. Second, our map-agnostic policy performs similarly to state-of-the-art adaptive IPP methods on various real-world terrain datasets. Third, with our experiments, we demonstrate that our map-agnostic adaptive IPP formulation easily integrates with and unifies previous non-learning-based state-of-the-art adaptive IPP algorithms while maintaining or improving their planning performance.

This chapter incorporates material from the following peer-reviewed journal publication, for which I have been the main contributor:

- Julius Rückin, David Morilla-Cabello, Cyrill Stachniss, Eduardo Montijano, and Marija Popović. Towards Map-Agnostic Policies for Adaptive Informative Path Planning. *IEEE Robotics and Automation Letters (RA-L)*, 10(5): 5114–5121, 2025

5.1 Map-Agnostic Adaptive Informative Path Planning

We aim to formulate the adaptive IPP problem for terrain monitoring [63, 64, 102, 106, 126] in a map-agnostic fashion to offline-learn or online-solve planning policies across different monitoring missions and map representations without redesigning or re-training policies. In Sec. 5.1.1, we start by making the challenge of map-agnostic adaptive IPP mathematically precise and introduce the associated sequential decision-making problem. Next, we present our novel map-agnostic formulation of the adaptive IPP problem in Sec. 5.1.2. Particularly, in Sec. 5.1.2.3, we show how to use this novel formulation to train a single map-agnostic adaptive IPP policy across a wide range of terrain monitoring missions.

5.1.1 Problem Formulation

We consider a robot with pose $\mathbf{p}_t \in \mathbb{R}^{D_r}$ at time t, moving in an a priori unknown terrain. The terrain $\xi \subset \mathbb{R}^{D_e}$ is characterised by its unknown and stationary feature field $F : \xi \to \mathcal{F}$. The continuous or discrete terrain feature space $\mathcal{F}$ is the mission-specific information the robot is tasked with to gather. The goal is to estimate and precisely map the terrain feature field F in interesting areas,

$$\xi_I = \{\mathbf{x} \in \xi \mid F(\mathbf{x}) \in \mathcal{F}_I\} \subseteq \xi\,, \tag{5.1}$$

where $\mathcal{F}_I \subseteq \mathcal{F}$ is the user-defined set of feature values qualifying a point $\mathbf{x} \in \xi$ as interesting, e.g. a subset of value ranges or a subset of semantic classes.

To accomplish this objective, the robot is equipped with a sensor to collect measurements $z \in \mathcal{Z}$ from the terrain, e.g. semantically segmented RGB images, thermal images, or radiation levels. At each time step t, the measurements provide noisy information about F according to $z_t \sim p(z \mid \mathbf{p}_t, F)$. They are used to model a stochastic process $\hat{F}_t$ over all possible terrain feature field functions F,

$$\hat{F}_t \sim p(F \mid z_{1:t}, \mathbf{p}_{1:t}, \theta_F)\,, \tag{5.2}$$

where $z_{1:t} \in \mathcal{Z}^t$ is the set of all measurements collected at poses $\mathbf{p}_{1:t}$. The map representation is indicated and parameterised by hyperparameters θ_F. Most works update the map belief $\hat{F}_t$ for continuous-valued features $\mathcal{F} \subseteq \mathbb{R}$ with pre-trained Gaussian processes or Kalman filters [177]. Map beliefs $\hat{F}_t$ for discrete-valued features $\mathcal{F} \subseteq \mathbb{N}$ are commonly updated using occupancy grid mapping [114].

We aim to find an optimal action sequence or path $\psi^* = (\mathbf{a}_1, \ldots, \mathbf{a}_N)$, where $\mathbf{a}_t \in \mathcal{A} \subseteq \mathbb{R}^{D_a}$ are relative pose changes. The path ψ^* maximises an information criterion $I : \mathcal{A}^N \times \xi_I \to \mathbb{R}$, where $\mathcal{A}^N$ encompasses all paths of length N. The information criterion, as in Eq. (4.1), associates measurements $z_{1:N}$ collected along path ψ with their information value $I(\psi, \xi_I)$ about areas of interest ξ_I, so that

$$\psi^* = \underset{\psi \in \mathcal{A}^N}{\operatorname{argmax}}\ I(\psi, \xi_I), \text{ s.t. } C(\psi) \leq B, \tag{5.3}$$

is the optimal path with the highest information value about areas of interest. The path's execution cost is given by $C : \mathcal{A}^N \to \mathbb{R}$, e.g. battery capacity or travel time. The robot's maximum mission budget is denoted as $B \geq 0$. As feature field F and thus areas of interest ξ_I are a priori unknown, Eq. (5.3) cannot be solved offline. The optimal path ψ^* in Eq. (5.3) changes as $\hat{F}_t$ is updated based on new measurements. Therefore, online replanning is required to find an optimal path ψ^* that adaptively focuses on areas of interest ξ_I as they are discovered.

The concrete formulation of Eq. (5.3) depends on the specific terrain monitoring mission. Depending on the mission, the spatially mapped terrain features $\mathcal{F}$ might be discrete, such as semantic classes, or continuous, such as surface temperature. Different terrain features might require different map representations $\hat{F}_t$ with hyperparameters θ_F. Furthermore, the user specifies feature values $\mathcal{F}_I \subseteq \mathcal{F}$ that qualify an area to be of interest. We denote $\mathcal{H} = \{\mathcal{F}, \mathcal{F}_I, \theta_F\}$ as the set of mission hyperparameters defining the specific instantiation of Eq. (5.3).

As shown in Sec. 4.1, the adaptive IPP problem in Eq. (5.3) can be transformed into a sequential decision-making problem solvable with RL by

$$\begin{aligned} \pi^* &= \underset{\pi \in \Pi}{\operatorname{argmax}}\, I\left((\pi(s_0), \ldots, \pi(s_{N-1})), \xi_I\right) \\ &= \underset{\pi \in \Pi}{\operatorname{argmax}} \sum_{t=0}^{N-1} \gamma^t R\left(s_t, \pi(s_t), s_{t+1}, \xi_I\right), \end{aligned} \tag{5.4}$$

where $\pi : \mathcal{S} \to \mathcal{A}$ is a planning policy mapping state $s_t \in \mathcal{S}$ at time step t to an action $\mathbf{a}_t = \pi(s_t)$, and Π is the function space of all possible policies. Thus, the path ψ is given by $\psi = (\pi(s_0), \dots, \pi(s_{N-1}))$. Usually, a mission- and map-specific reward function $R(s_t, \pi(s_t), s_{t+1}, \xi_I) \in \mathbb{R}$ resembles the information criterion I. It rewards taking actions $\mathbf{a}_t$ in state s_t that lead to a next state s_{t+1} with increased information about areas of interest ξ_I. The discount factor $\gamma \in [0, 1]$ weighs the importance of future rewards. As areas of interest ξ_I are unknown, prior adaptive IPP methods [19, 61, 126, 135, 175] approximate unknown areas of interest ξ_I. To this end, they use map-specifically computed confidence intervals based on hand-tuned confidence thresholds, e.g. as shown in Eq. (4.3), rewarding uncertainty reduction over map belief $\hat{F}_t$ in these approximated areas of interest.

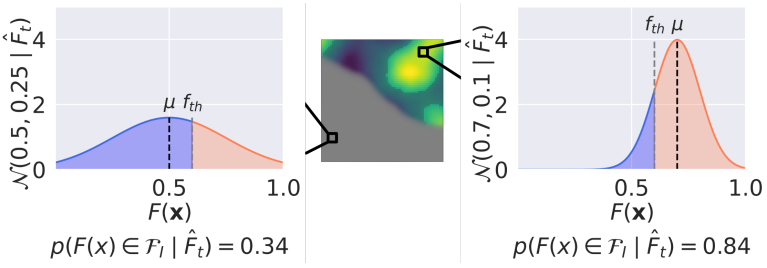
Different from existing adaptive IPP approaches that consider terrain map-specific planning state formulations s_t with approximated areas of interest, in Sec. 5.1.2, we formulate the problem in Eq. (5.4) in a fully probabilistic and map-agnostic fashion. To this end, we propose a planning state s_t that unifies the adaptive IPP problem across different map representations $\hat{F}_t$. This allows us to apply a single learned policy π^* to varying terrain monitoring missions. Based on this planning state, we introduce a new reward function for Eq. (5.4) enabling training or online-solving policy π^* for different terrain monitoring missions. Additionally, we use the mission hyperparameters $\mathcal{H}$ to condition the learned policy π^* on the user-defined and mission-specific problem characteristics.

5.1.2 Map-Agnostic Planning Policy

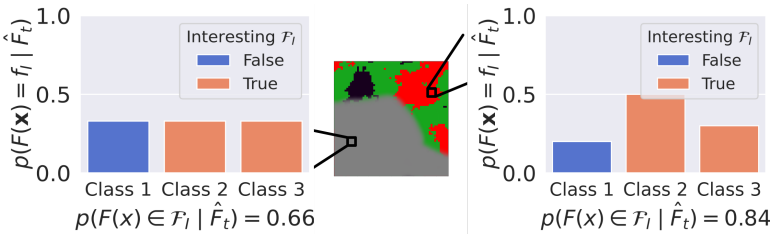
Our approach is conceptually depicted in Fig. 5.1. We unify the adaptive IPP problem formalised in Sec. 5.1.1 across different map representations required to spatially capture continuous- and discrete-valued terrain features. To this end, we view any terrain monitoring mission as a binary segmentation task, probabilistically splitting the terrain into areas of interest and uninteresting areas. Based on this belief over areas of interest, we propose a map-agnostic planning state space in Sec. 5.1.2.1 and introduce a new reward function for adaptive IPP in Sec. 5.1.2.2. This allows us to online-solve or offline-train a planning policy in a unified fashion across different map representations. Last, in Sec. 5.1.2.3, we show how we use our state space and reward function to offline-train an adaptive IPP policy on varying terrain monitoring missions in simulation.

5.1.2.1 Unified Planning State Space

Our formulation of planning states $s_t \in \mathcal{S}$ encodes all information required to solve the adaptive IPP problem in Eq. (5.4). This includes the robot's state estimation, its current understanding of the terrain, and mission hyperparameters.



(a) Continuous-valued terrain features



(b) Discrete-valued terrain features

Figure 5.2: Our unified belief $p\big(F(\mathbf{x}) \in \mathcal{F}_I \mid \hat{F}_t\big)$ over areas of interest ξ_I for continuous- and discrete-valued terrain features. (a) Posterior normal distributions inferred from a Gaussian process or Kalman filter map representation at a point $\mathbf{x}$ in the terrain ξ with an interesting value threshold $f_{th} = 0.6$. The unified belief is computed by the orange area under the curve, which is larger for observed areas of interest (yellow) than for unknown uncertain areas (grey). (b) The unified belief is given by the sum of posterior probability masses over interesting classes (orange) with higher probability for observed areas of interest (green, red). Posterior probability masses are extracted from occupancy grid map cells corresponding to point $\mathbf{x}$ in the terrain ξ.

We propose a belief over areas of interest ξ_I as input to the planning policy $\pi(s_t)$ that unifies continuous- and discrete-valued terrain features requiring different map representations $\hat{F}_t$ in Eq. (5.2). Assume that $\mathcal{X}_t$ is a finite subset of points $\mathbf{x}_t \in \xi$ spatially sampled from the terrain ξ at time step t at which we aim to infer the state $s_t(\mathbf{x}_t)$. Then, for each point $\mathbf{x}_t \in \mathcal{X}_t$, the state $s_t(\mathbf{x}_t)$ is defined as

$$s_t(\mathbf{x}_t) = \Big(p\big(F(\mathbf{x}_t) \in \mathcal{F}_I \mid \hat{F}_t\big),\, H\big(\hat{F}_t(\mathbf{x}_t)\big),\, \mathbf{p}_t,\, B_t,\, \mathcal{H}\Big)\,, \tag{5.5}$$

where $p\big(F(\mathbf{x}_t) \in \mathcal{F}_I \mid \hat{F}_t\big)$ is the probability of $\mathbf{x}_t$ being part of an area of interest ξ_I, $H\big(\hat{F}_t(\mathbf{x}_t)\big)$ is the uncertainty of the mission-specific map belief $\hat{F}_t$ at point $\mathbf{x}_t$, $\mathbf{p}_t$ is the robot's current position, $B_t \leq B$ is the remaining budget, and $\mathcal{H}$ are the mission hyperparameters specifying Eq. (5.3). For occupancy maps, $H\big(\hat{F}_t(\mathbf{x}_t)\big)$ is the Shannon entropy at $\mathbf{x}_t$. For Gaussian processes or Kalman filters, $H\big(\hat{F}_t(\mathbf{x}_t)\big)$ is the variance at $\mathbf{x}_t$. Our method supports any spatial arrangement of points $\mathcal{X}_t$ at which we capture the planning state. It can be integrated with any equidistant grid-like sampling of the state space over the terrain ξ as used in [117, 126, 135, 175] or topological sampling as used in [18, 19, 167, 174].

In contrast to previous works relying on map-specific formulations of state s_t with binary approximations of interesting areas, our planning state formulation in

Eq. (5.5) introduces a fully probabilistic and map-agnostic belief over areas of interest, denoted as $\hat{F}_{I,t} \sim p\big(F(\mathbf{x}_t) \in \mathcal{F}_I \mid \hat{F}_t\big)$. Next, we show how to compute this unified belief for continuous- and discrete-valued terrain feature mapping missions with different map representations and illustrate its computation in Fig. 5.2.

Consider discrete feature spaces $\mathcal{F} = \{1, \dots, K\}$ with $K \in \mathbb{N}$ semantic classes. Areas of interest ξ_I are given by a user-defined set of interesting classes $\mathcal{F}_I \subseteq \mathcal{F}$ with $|\mathcal{F}_I| \leq K$. As the map belief $\hat{F}_t \sim p(F \mid z_{1:t}, \mathbf{p}_{1:t})$ is represented using occupancy grid maps, the unified belief $\hat{F}_{I,t}(\mathbf{x})$ over areas of interest is defined as

$$p\big(F(\mathbf{x}) \in \mathcal{F}_I \mid \hat{F}_t\big) = \sum_{f_I \in \mathcal{F}_\mathcal{I}} p\big(F(\mathbf{x}) = f_I \mid z_{1:t}, \mathbf{p}_{1:t}\big)\,, \tag{5.6}$$

where f_I is a single class in the set of interesting classes $\mathcal{F}_I$. The f_I-th layer of the occupancy map at the grid cell corresponding to $\mathbf{x} \in \xi$ provides the probability $p\big(F(\mathbf{x}) = f_I \mid z_{1:t}, \mathbf{p}_{1:t}\big)$ of point $\mathbf{x}$ belonging to class f_I.

Next, consider continuous feature spaces $\mathcal{F} = [f_a, f_b]$ with $f_a \leq f_b$. Interesting areas are given by user-defined thresholds f_{th} with $f_a \leq f_{th} \leq f_b$, such that $\mathcal{F}_I = [f_{th}, f_b]$. As the map belief $\hat{F}_t$ is represented by a Gaussian process or Kalman filter, the probability density over feature values is given by $\hat{F}_t \sim \mathcal{N}(\mu(\mathbf{x}), \sigma(\mathbf{x})^2 \mid \hat{F}_t)$ with posterior mean $\mu_t(\mathbf{x})$ and variance $\sigma_t(\mathbf{x})^2$ of $\hat{F}_t$ at point $\mathbf{x}$. The unified belief $p\big(F(\mathbf{x}) \in \mathcal{F}_I \mid \hat{F}_t\big)$ over interesting areas is defined as

$$\begin{aligned} p\big(F(\mathbf{x}) \in \mathcal{F}_I \mid \hat{F}_t\big) &= \frac{1}{\sqrt{2\pi\sigma_t(\mathbf{x})^2}} \int_{f_{th}} \exp\left(-\frac{(f - \mu_t(\mathbf{x}))^2}{2\sigma_t(\mathbf{x})^2}\right) df \\ &= 1 - \Phi\left(\frac{f_{th} - \mu_t(\mathbf{x})}{\sqrt{\sigma_t(\mathbf{x})^2}}\right), \end{aligned} \tag{5.7}$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution measuring $p\big(F(\mathbf{x}) \leq f_{th} \mid \hat{F}_t\big)$, integrating w.r.t. feature values $f \geq f_{th}$.

The mission-specific hyperparameters $\mathcal{H} = \{\mathcal{F}, \mathcal{F}_I, \theta_F\}$ directly influence the computation of our unified belief over interesting areas $\hat{F}_{I,t}$ in Eq. (5.6) and Eq. (5.7). This makes the effect of the chosen mission hyperparameters accessible to the planning policy, thus improving adaptivity to the concrete instance of Eq. (5.4) a planning method aims to solve. For learning-based planning methods aiming to train a policy π^* offline, we additionally condition the planning policy on the mission-specific hyperparameters as it allows us to train a single policy that can solve Eq. (5.4) for various terrain monitoring variants $\mathcal{H}$ without retraining.

5.1.2.2 Adaptive Planning Reward Function

We introduce a new reward function for the adaptive IPP problem formalised in Eq. (5.4) based on our unified planning state formulation s_t presented in Sec. 5.1.2.1. The unified planning state formulation and reward function could be

integrated into any non-learning-based planning method searching for the optimal policy π^* online or learning-based planning method for training π^* offline.

Our goal is to quickly find areas of interest ξ_I in Eq. (5.1) and precisely estimate the feature field F in these areas. Hence, we aim to maximise information about the map belief in areas of interest. To adapt paths towards areas likely of interest, we reward uncertainty reduction in $\hat{F}_t$ proportionally to our belief over interesting areas $\hat{F}_{I,t}(\mathbf{x}) \sim p\big(F(\mathbf{x}) \in \mathcal{F}_I \mid \hat{F}_t\big)$ as defined in Eq. (5.6) and Eq. (5.7). Assume $\mathcal{X}$ is a finite subset of points $\mathbf{x} \in \xi$ representing an equidistant grid over the terrain. We define the reward function in Eq. (5.4) as

$$R(s_t, \mathbf{a}_t, s_{t+1}) = \sum_{\mathbf{x} \in \mathcal{X}} \frac{H\big(\hat{F}_t(\mathbf{x})\big) - H\big(\hat{F}_{t+1}(\mathbf{x})\big)}{H\big(\hat{F}_t(\mathbf{x})\big)} p\big(F(\mathbf{x}) \in \mathcal{F}_I \mid \hat{F}_t\big), \tag{5.8}$$

where $H\big(\hat{F}_t(\mathbf{x})\big)$ is the uncertainty of the mission-specific map belief $\hat{F}_t$ at a point $\mathbf{x} \in \mathcal{X}$. The map belief $\hat{F}_{t+1}$ at the next time step $t+1$ is updated according to Eq. (5.2) after executing action $\mathbf{a}_t$ and collecting a new sensor measurement $z_{t+1} \sim p(z \mid \mathbf{p}_{t+1}, F)$ from a next pose $\mathbf{p}_{t+1}$. For occupancy maps, $H\big(\hat{F}_t(\mathbf{x}_t)\big)$ is the exponential Shannon entropy at $\mathbf{x}$. For Gaussian processes and Kalman filters, $H\big(\hat{F}_t(\mathbf{x}_t)\big)$ is the variance at $\mathbf{x}$.

Rewarding the uncertainty reduction of our new unified belief over areas of interest $\hat{F}_{I,t}$ would be sufficient to achieve competitive adaptive IPP performance in many terrain monitoring missions. However, for maximal performance across various missions, it is necessary to reward uncertainty reduction of the mission-specific map belief $\hat{F}_t$ in areas likely of interest as in Eq. (5.8). As an example, consider exploration missions with interesting features $\mathcal{F}_I = \mathcal{F}$. Then, interesting areas $\xi_I = \xi$ cover the whole terrain by definition of Eq. (5.1). Assuming occupancy maps, by definition of Eq. (5.6), for any point $\mathbf{x} \in \xi$, it holds $p(F(\mathbf{x}) \in \mathcal{F}_I \mid \hat{F}_t) = 1$ even before a mission starts. Hence, the uncertainty of the unified belief over areas of interest is $H(\hat{F}_{I,t}(\mathbf{x})) = 0$, thus not providing informative rewards to map the terrain feature field F precisely. Instead, we reward uncertainty reduction over the map belief $\hat{F}_t$ proportionally to our new unified belief over interesting areas $\hat{F}_{I,t}$. In this way, our reward function encourages exploring uncertain areas of the terrain while adaptively refining the map belief in areas of interest as they are discovered during a mission.

5.1.2.3 Planning Policy Training Details

We use RL to train a single unified planning policy π^* on simulated terrain monitoring deployments with previously unmet mission variations. We detail our terrain monitoring mission simulations, encoding of mission hyperparameters $\mathcal{H}$ and the used RL algorithm and policy network representing π^*. In practice, any policy learning method, e.g. imitation learning, policy network architecture, and

hyperparameter encoding could be used to train the policy with our map-agnostic adaptive IPP formulation introduced in Sec. 5.1.2.1 and Sec. 5.1.2.2.

Mission simulations. We randomly choose missions that aim to map terrain feature fields F either with continuous- or with discrete-valued features $\mathcal{F}$. In the case of continuous-valued features, we use a Gaussian process with randomly sampled kernel hyperparameters θ_F to represent and update the map belief $\hat{F}_t$ in Eq. (5.2). For discrete-valued features, we use an occupancy map to represent and update $\hat{F}_t$. We simulate randomised ground truth feature fields F with spatial correlations of different extents, as depicted in Fig. 5.4 to encourage learning planning strategies generalising to varying feature fields. We randomly sample the set of interesting feature values $\mathcal{F}_I$ from the feature space $\mathcal{F}$.

Hyperparameter encoding. We explicitly input mission hyperparameters l_{GP} and f_{th} into state s_t. The map hyperparameter $l_{GP} \geq 0 \in \theta_F$ is the lengthscale of a Gaussian process Matern kernel used to represent the map belief $\hat{F}_t$. This is important as different lengthscales result in different map updates along paths, potentially affecting decision-making. Map beliefs $\hat{F}_t$ assuming spatially independent measurements z, e.g. occupancy grid maps, are naturally encoded by $l_{GP} = 0$ as Matern kernels with $l_{GP} \to 0$ assume spatially independent measurements. The user-defined value $f_{th} \in \mathcal{F}$ determines the interesting features $\mathcal{F}_I$. Interesting feature thresholds f_{th} encode prior user belief about the spatial extent of interesting areas ξ_I over the terrain ξ. For example, for continuous- or discrete-valued missions with $\mathcal{F}_I = \mathcal{F}$, conditioning the policy on $f_{th} = 0$ encodes those interesting areas that cover the entire terrain.

Policy training. We train our policy π^* using proximal policy optimisation (PPO) [146], which is an actor-critic RL algorithm, as it is known for its versatile usage and reasonable training stability. For details on actor-critic RL algorithms, we refer to Chap. 3. We compute our planning state representation in Eq. (5.5) over an equidistant grid $\mathcal{X}_t$. Our shared actor-critic neural network is parameterised by θ and processes this state representation as shown in Fig. 5.3.

We use the IMPALA encoder [41] to process the unified belief over areas of interest $p\big(F(\mathbf{x}) \in \mathcal{F}_I \mid \hat{F}_t\big)$ and map belief uncertainty $H\big(\hat{F}_t(\mathbf{x})\big)$. The IMPALA encoder consists of three convolutional blocks with $16, 32$ and 32 channels, respectively. Each block consists of a convolutional layer followed by max-pooling to downsample feature maps by a factor of 2 and two subsequent residual blocks as proposed by He et al. [59]. In total, the IMPALA encoder consists of 15 convolutional layers. After the last convolutional layer, a fully connected non-linear layer projects the feature maps to a 256-dimensional latent map state vector.

We use a multilayer perceptron (MLP) to process the current robot's position $\mathbf{p}_t$, remaining budget B_t and mission hyperparameters $\mathcal{H}$. These state values are encoded using positional encoding [168]. Next, we process them with the

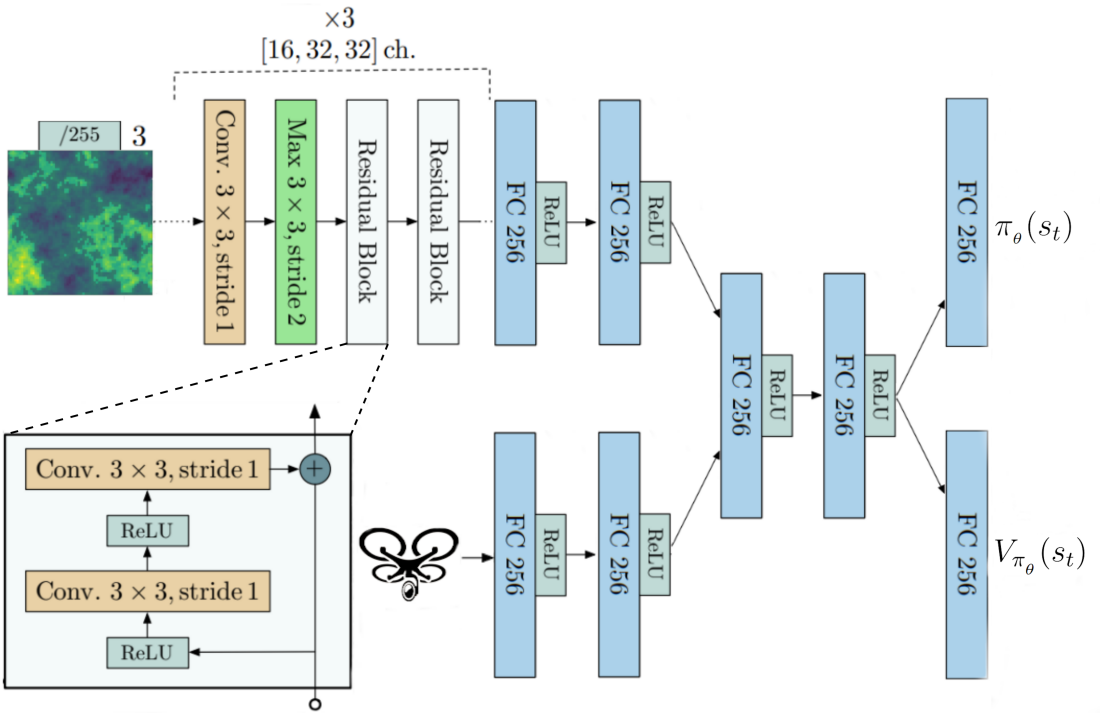


Figure 5.3: Our shared actor-critic neural network architecture used by the proximal policy optimisation algorithm to process planning state information. We use the IMPALA encoder [41] combined with a simple two-layer MLP encoder to process map and robot state information, respectively. Based on the shared latent state vector, individual actor and critic predictions heads output the stochastic policy $\pi_\theta(s_t)$ and state-value function $V_{\pi_\theta}(s_t)$ given state s_t.

2-layer MLP with 256 hidden units per layer to output the latent robot state vector. Subsequently, the latent map and robot state vectors are concatenated and processed by another 2-layer MLP with 256 hidden units per layer to output a latent state vector. Based on this latent state vector, two independent prediction heads, consisting of a single linear layer each, output the stochastic policy $\pi_\theta(s_t) \in [0,1]^{|\mathcal{A}|}$ and state-value function $V_{\pi_\theta}(s_t) \in \mathbb{R}$, where $|\mathcal{A}|$ is the number of actions in action space $\mathcal{A}$. We train the network until performance converges.

During deployment, we perform inference based on the policy π^*_θ parameterised by the trained actor-critic network θ. At each time step t, we select the next action $\mathbf{a}_t \in \mathcal{A}$ with the highest probability based on state s_t as

$$\mathbf{a}_t = \underset{i \in \{1,\dots,|\mathcal{A}|\}}{\operatorname{argmax}} \pi^*_\theta(s_t)_i. \tag{5.9}$$

Note that our proposed method could be combined with any other RL algorithm to train adaptive IPP policies and any other actor-critic network architecture to process state information computed at $\mathcal{X}_t$. Advances in network architectures designed for learning-based adaptive IPP methods, such as the one introduced by Cao et al. [19], are orthogonal research directions to our method.

5.2 Experimental Evaluation

The experiments are designed to evaluate our approach and experimentally back up the claims made in this chapter. First, in Sec. 5.2.2, we show that training our single unified policy on various monitoring missions yields competitive performance with state-of-the-art online non-learning-based policy search methods and offline-learned policies adapted and re-trained for each class of monitoring missions with specific map representations. Second, in Sec. 5.2.3, we verify that our single unified policy trained in simulation performs similarly to these state-of-the-art adaptive IPP methods on unseen real-world datasets. Finally, in Sec. 5.2.4, we demonstrate that our map-agnostic adaptive IPP formulation unifies existing adaptive IPP methods while maintaining or improving planning performance.

5.2.1 Experimental Setup

Mission setup. The procedure for simulating monitoring missions is described in Sec. 5.1.2.3. For discrete-valued terrain features, we assume three classes $\mathcal{F}$ with interesting classes $\mathcal{F}_I$ of varying spatial extent. We equip a simulated UAV with a sensor delivering image-like semantic measurements z_t spanning a downwards-projected field of view. We use occupancy grid maps $\hat{F}_t$ for terrain mapping and confusion matrix-based sensor noise as in [126, 175]. For continuous-valued terrain features, we assume features $\mathcal{F} = [0, 1]$ with interesting thresholds f_{th}, such that $\mathcal{F}_I = [f_{th}, 1]$. Simulated UAVs are equipped with sensors delivering point measurements z_t with Gaussian noise, mapped using Gaussian processes as in [19, 61, 117, 126]. We distinguish between the classical evaluation on fixed mission hyperparameters $\mathcal{H} = \{f_{th}, l_{GP}\} = \{0.4, 0.35\}$ as in [19, 61, 126, 135, 175], denoted as static, and our more challenging scenario of randomly sampled $\mathcal{H}$ with $f_{th} \in [0.0, 0.8]$ and $l_{GP} \in [0.15, 0.55]$ denoted as varying. The mission budget is set to $B = 100$s, and initial robot positions $\mathbf{p}_0$ are sampled randomly. We assume relative 2D robot position changes $\mathbf{a}_t \in \mathcal{A}$ on an equidistant grid as in [117, 126, 135, 175]. To benchmark our approach, we simulate ground truth feature fields F with varying spatial correlations as shown in Fig. 5.4. Additionally, we evaluate on real-world orthomosaic feature fields F in Fig. 5.5.

Baselines. We consider state-of-the-art adaptive IPP methods as baselines. In contrast to our RL-Ours method, all baseline methods rely on map-specific planning state spaces. All baseline methods consider the current robot position $\mathbf{p}_t$ and remaining budget B_t in their state. Continuous-valued terrain features are modelled by directly using posterior mean $\mu_t(\mathbf{x})$ and variance $\sigma_t(\mathbf{x})^2$ of the Gaussian process at points $\mathbf{x} \in \mathcal{X}_t$ as in [19, 61, 117, 126, 135]. Discrete-valued terrain features are modelled by directly using the posterior occupancy map $p\big(F(\mathbf{x}) \mid z_{1:t}, \mathbf{p}_{1:t}\big)$ and its entropy $H(\hat{F}_t(\mathbf{x}))$ at points $\mathbf{x} \in \mathcal{X}_t$ as in [126, 175]. All

Table 5.1: Hyperparameters of (left) Monte Carlo tree search (MCTS) and (right) covariance matrix adaptation evolution strategy (CMA-ES) used as online policy search algorithms.

	Hyperparameter	Value		Hyperparameter	Value
MCTS	Horizon (T)	5	**CMA-ES**	Horizon (T)	5
	Discount factor (γ)	1.0		Discount factor (γ)	1.0
	Num. simulations	65		Max. iterations	10
	Exploration coeff.	1.41		Population size	8
	Rollout policy	Uniform		Initial std. dev.	0.08

baseline methods reward map uncertainty reduction in areas of interest approximated by hand-tuned and map-specific confidence intervals as in [19, 61, 126].

Based on these map-specific states and rewards, we implement state-of-the-art online policy search methods and offline-train planning policies using RL. As uncertainty reduction depends on unknown future sensor measurements, for online policy search methods, at time step t, we simulate expected future sensor measurements z_{t+1}. To this end, during online planning, we sample z_{t+1} from the current probabilistic posterior map belief $\hat{F}_t$ according to $z_{t+1} \sim p(z \,|\, \mathbf{p}_{t+1}, \hat{F}_t)$ at a potential future robot pose $\mathbf{p}_{t+1}$. We use the respective inverse sensor model to transform z_{t+1} into a probabilistic observation to update the posterior map $\hat{F}_{t+1}$.

As online policy search methods, we implement (i) MCTS, a rollout-based solver to plan the next action based on sampled future finite-horizon paths to estimate next actions' value functions, similar to Ott et al. [117]; (ii) CMA-ES fine-tunes an initial greedily chosen path over a finite horizon in continuous space over the terrain in an evolutionary fashion using the covariance matrix adaptation evolution strategy (CMA-ES) [57] as proposed by Popović et al. [126]; (iii) Greedy executes a greedily chosen next-best action as described by Popović et al. [126]. Hyperparameters for MCTS and CMA-ES are summarised in Tab. 5.1 and tuned for performance while avoiding excessive replanning run times.

To offline-train RL-Base planning policies, we use RL assuming Static hyperparameters and perform policy inference online as in [18, 19, 116, 167, 174, 175]. For a fair comparison with our novel map-agnostic RL-Ours policy, we use the same actor-critic network trained with PPO for all RL-Base policies as detailed in Sec. 5.1.2.3. PPO hyperparameters used to train all learning-based policies are summarised in Tab. 5.2. Furthermore, we pre-compute lawnmower-like coverage paths commonly used in real-world terrain monitoring deployments.

Evaluation metrics. All adaptive replanning performance metrics are computed over areas of interest ξ_I in Eq. (5.1) after a mission is terminated. For continuous-valued mapping missions, we compute the final covariance log-trace of map $\hat{F}_t$ normalised by the prior covariance log-trace of $\hat{F}_0$ (Unc.), and RMSE and mean log loss (MLL) of $\hat{F}_t$ w.r.t. the ground truth feature field F as in [61, 102, 126]. For discrete-valued mapping missions, we compute the final

Table 5.2: PPO hyperparameters used to train RL-Ours, RL-Base-C, and RL-Base-D policies.

Hyperparameter	**Value**
Horizon (T)	64
Optimizer	Adam with $\epsilon = 10^{-5}$
Learning rate	LinearAnneal$(3 \cdot 10^{-4}, 0)$
Num. epochs	8
Minibatch size	4096
Num. workers	256
Discount factor (γ)	0.99
GAE factor (λ)	0.95
Entropy coefficient	0.01
Clipping range	0.2
VF coefficient	1.0
Max. grad. norm	0.5
Num. env. steps	10^7
Map state dimensions $\mathcal{X}_t$	64×64

Shannon entropy of map $\hat{F}_t$ normalised by the prior Shannon entropy of $\hat{F}_0$ (Unc.), and mean Intersection-over-Union (mIoU) and F1-score of $\hat{F}_t$ w.r.t. the ground truth feature field F as in [116, 126, 175]. Moreover, we compute an information integral (II) as one minus the area under the normalised map uncertainty (Unc.) over budget curve. The II captures the uncertainty reduction speed over the depleted budget in a single metric. All metrics are averaged over 100 missions, repeated with three different random seeds. We report mean and standard deviations over the three seeds. Each metric is formalised as follows.

Assume $\mathcal{X}$ is a subset of points $\mathbf{x} \in \xi$ sampled from an equidistant grid over the terrain ξ. First, we show how to compute the evaluation metrics used to measure performance in the continuous-valued terrain monitoring missions.

The RMSE captures the error between the map belief $\hat{F}_t$ and the ground truth feature field F in interesting areas ξ_I,

$$\mathrm{RMSE}\big(\hat{F}_t, F, \xi_I\big) = \sqrt{\frac{1}{N_I} \sum_{\mathbf{x} \in \mathcal{X}} \mathbb{I}_{\mathbf{x} \in \xi_I} \left(\mu_t(\mathbf{x}) - F(\mathbf{x})\right)^2}\,, \tag{5.10}$$

where $t = 100$ as we are interested in the map's final RMSE, $\mu_t(\mathbf{x})$ is the posterior mean of map belief $\hat{F}_t$ at point $\mathbf{x}$, and $N_I = |\{\mathbf{x} \in \mathcal{X} \mid \mathbf{x} \in \xi_I\}|$ is the number of points in $\mathcal{X}$ that belong to areas of interest ξ_I. The indicator variable $\mathbb{I}_{\mathbf{x} \in \xi_I}$ is equal to 1, if $\mathbf{x}$ is part of an area of interest ξ_I, else 0.

The MLL captures the uncertainty-weighted error between the map belief $\hat{F}_t$ and the ground truth feature field F in interesting areas ξ_I,

$$\mathrm{MLL}\big(\hat{F}_t, F, \xi_I\big) = \frac{1}{N_I} \sum_{\mathbf{x} \in \mathcal{X}} \mathbb{I}_{\mathbf{x} \in \xi_I} \left(\frac{1}{2} \log(2\pi\sigma_t(\mathbf{x})^2) + \frac{(\mu_t(\mathbf{x}) - F(\mathbf{x}))^2}{2\sigma_t(\mathbf{x})^2}\right), \tag{5.11}$$

where $t = 100$ as we are interested in the map's final MLL, and $\sigma_t(\mathbf{x})^2$ is the posterior variance of map belief $\hat{F}_t$ at point $\mathbf{x}$.

The remaining uncertainty of the map belief $\hat{F}_t$ in areas of interest ξ_I is

$$\mathrm{Unc}\big(\hat{F}_t, \xi_I\big) = \frac{1}{N_I} \sum_{\mathbf{x} \in \mathcal{X}} \mathbb{I}_{\mathbf{x} \in \xi_I} \frac{\log(\sigma_t(\mathbf{x})^2)}{\log(\sigma_0(\mathbf{x})^2)}\,, \tag{5.12}$$

where $t = 100$ as we are interested in the map's final uncertainty, and $\sigma_0(\mathbf{x})^2$ is the map belief's prior variance at $\mathbf{x}$. In this way, we compute the Gaussian process covariance matrix trace. As the covariance trace is reduced exponentially with the number of collected sensor measurements and since the overall magnitude depends on the Gaussian process prior variance, we scale the posterior variances logarithmically and normalise by the map's prior log-variance.

Second, we show how to compute the evaluation metrics used to measure performance in the discrete-valued terrain monitoring missions. We compute the confusion matrix between map predictions $\mathrm{argmax}_{k \leq K}\, p(F(\mathbf{x}) = k \mid \hat{F}_t)$ and ground truth $F(\mathbf{x})$ for all $\mathbf{x} \in \xi_I$, where K is the number of classes. Then, the mIoU captures the error between the map predictions and the ground truth feature field F in areas of interest ξ_I,

$$\mathrm{mIoU}\big(\hat{F}_t, F, \xi_I\big) = \frac{1}{|\mathcal{F}_I|} \sum_{k \in \mathcal{F}_I} \frac{TP(k)}{TP(k) + FP(k) + FN(k)}\,, \tag{5.13}$$

where $t = 100$ as we are interested in the map's final mIoU. The true and false positives $TP(k)$ and $FP(k)$, and true and false negatives $TN(k)$ and $FN(k)$ for a class k in the set of interesting classes $\mathcal{F}_I$ are given by the confusion matrix.

Similarly, the F1-score captures the error between the map belief $\hat{F}_t$ and the ground truth feature field F in areas of interest ξ_I,

$$\mathrm{F1}\big(\hat{F}_t, F, \xi_I\big) = \frac{1}{|\mathcal{F}_I|} \sum_{k \in \mathcal{F}_I} \frac{2\,TP(k)}{2\,TP(k) + FP(k) + FN(k)}\,, \tag{5.14}$$

where $t = 100$ as we are interested in the map's final F1-score.

The remaining uncertainty of the map $\hat{F}_t$ in areas of interest ξ_I is defined as

$$\mathrm{Unc}\big(\hat{F}_t, \xi_I\big) = \frac{1}{N_I} \sum_{\mathbf{x} \in \mathcal{X}} \mathbb{I}_{\mathbf{x} \in \xi_I} \frac{H\big(\hat{F}_t(\mathbf{x})\big)}{H\big(\hat{F}_0(\mathbf{x})\big)}\,, \tag{5.15}$$

where $t = 100$ as we are interested in the map's final uncertainty, $H(\hat{F}_t(\mathbf{x}))$ is the Shannon entropy of the occupancy grid map's categorical distribution over all classes at the grid cell corresponding to $\mathbf{x}$, and $H_0(\mathbf{x})$ is the prior map belief's Shannon entropy at $\mathbf{x}$. As the posterior entropy's overall magnitude depends on the prior map entropy, we normalise it by the prior map's entropy.

Finally, the II of map belief $\hat{F}_t$ in areas of interest ξ_I is defined as

$$\mathrm{II}\big(\hat{F}_t, \xi_I\big) = 1 - \int_0^1 \mathrm{Unc}\big(\hat{F}_{t' \cdot B}, \xi_i\big)\, dt' \,, \tag{5.16}$$

where $t' = \frac{t}{B} \in [0, 1]$ with $t \leq B$ is the mission time step normalised by the initial mission budget B. Like this, the specified integral is the normalised area under the uncertainty-over-budget curve. This area captures how fast map uncertainty is reduced in areas of interest ξ_I. The area is bounded between 0 and 1 as time steps t' and uncertainty's $\mathrm{Unc}\big(\hat{F}_{t' \cdot B}\big)$ are normalised between 0 and 1. We assume a piecewise linear uncertainty-over-budget curve for numerical integration.

Hardware. We train the RL policies on a workstation equipped with an Intel i9 − 10940X 3.30GHz 14-core CPU, 64GB memory, and an NVIDIA RTX A5000 GPU. Training for 10^7 environment steps using the IMPALA-MLP policy network takes around 56 hours wall-clock time using our implementation. For the run time analysis of RL-based and online policy search approaches, we use a less powerful computer equipped with an Intel i7 − 1165G7 2.80GHz 4-core CPU, 32GB memory, and no built-in GPU. In this way, we closely mimic mobile robot compute constraints and avoid the unfair advantages of GPU-accelerated policy network inference over classical non-GPU-optimised planning algorithms.

5.2.2 Simulation Results

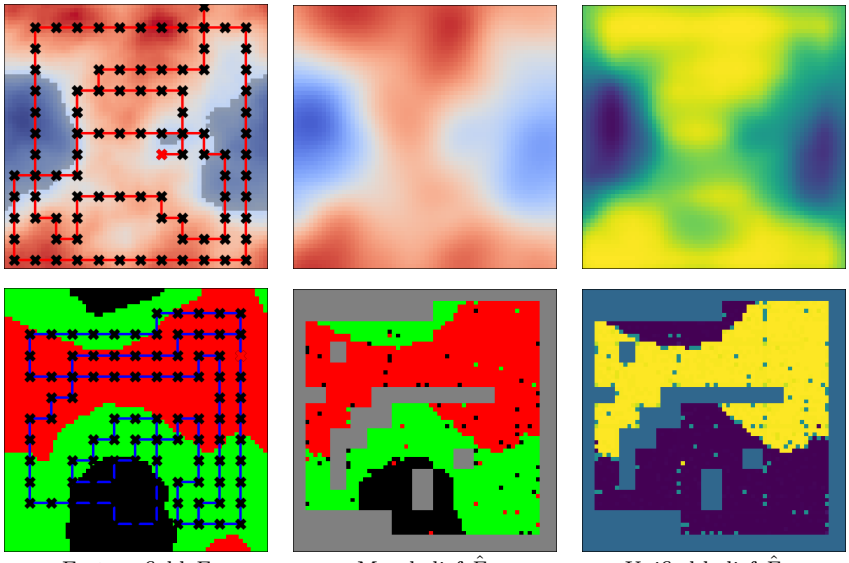
The first set of experiments investigates our first claim. We show that our map-agnostic adaptive IPP policy yields competitive performance with state-of-the-art online policy search methods while substantially reducing replanning runtime. Furthermore, our map-agnostic policy outperforms state-of-the-art map-specifically designed and offline-trained policies in varying terrain monitoring missions. We evaluate all methods in simulated continuous- and discrete-valued terrain feature monitoring scenarios as described in Sec. 5.2.1. We consider the classical static and our varying mission hyperparameter evaluation protocols to benchmark adaptive IPP approaches on challenging inter-mission variations.

Tab. 5.3 summarises the simulation results. In line with our findings in Chap. 4 and other RL-based adaptive IPP works, map-specifically designed and trained RL-Base-C and RL-Base-D policies outperform state-of-the-art online policy search methods in their respective continuous- and discrete-valued terrain feature monitoring missions with static hyperparameters they were trained on. Although we do not only train on missions with static hyperparameters, our single map-agnostic RL-Ours policy shows competitive performance on continuous- and discrete-valued monitoring missions with static hyperparameters. We achieve similar planning performance compared to online policy search methods and the RL-Base-C/D policies in these static scenarios.

Table 5.3: Comparison of state-of-the-art map-specifically designed and trained methods to our map-agnostic planning policy (RL-Ours) on simulated continuous- and discrete-valued terrain feature monitoring missions. Best average performances are marked in bold, and second-best average performances are underlined if standard deviations in brackets overlap. Our map-agnostic policy performs best in case of varying user-defined mission hyperparameters and similar to state-of-the-art adaptive IPP methods in case of static mission hyperparameters.

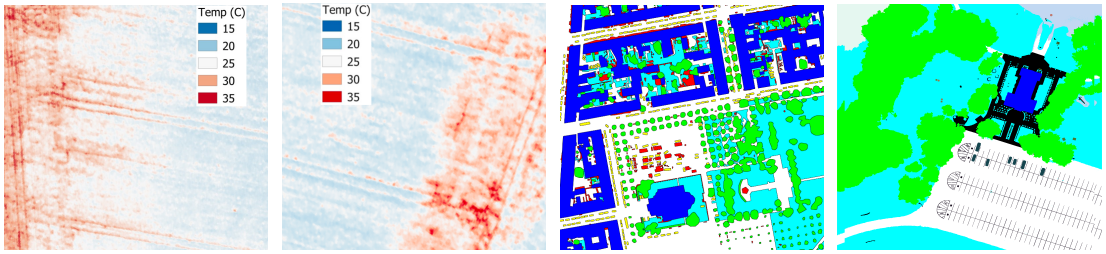
	Approach	**Static $\mathcal{H}$**				**Varying $\mathcal{H}$**				**Time**
		II↑	Unc.↓	MLL↓	RMSE↓	II↑	Unc.↓	MLL↓	RMSE↓	[s]↓
Continuous	RL-Ours	25.8 (0.17)	60.6 (0.22)	-64.6 (0.12)	3.83 (0.08)	**26.2** (0.65)	**60.4** (0.64)	**-60.5** (0.59)	3.81 (0.07)	**0.004**
	RL-Base-C	**26.1** (0.25)	**59.6** (0.28)	**-66.2** (0.39)	**3.67** (0.02)	24.0 (0.94)	64.2 (1.07)	-48.4 (4.64)	5.51 (0.97)	**0.004**
	MCTS	25.6 (0.09)	60.7 (0.08)	-64.9 (0.52)	3.83 (0.16)	25.3 (0.41)	61.1 (0.24)	-59.5 (0.70)	4.27 (0.22)	2.86
	CMA-ES	23.1 (1.27)	63.0 (3.41)	-60.1 (6.38)	5.45 (2.67)	21.5 (1.44)	64.3 (2.32)	-55.4 (5.52)	**2.59** (0.85)	6.05
	Greedy	24.5 (0.14)	62.0 (0.12)	-62.0 (0.26)	4.11 (0.11)	25.2 (0.66)	61.7 (0.29)	-58.0 (1.55)	4.35 (0.24)	0.05
	Coverage	15.3 (0.16)	75.5 (0.39)	-28.0 (1.44)	10.8 (0.12)	13.4 (0.28)	77.1 (0.25)	-30.6 (0.87)	9.93 (0.20)	-

	Approach	**Static $\mathcal{H}$**				**Varying $\mathcal{H}$**				**Time**
		II↑	Unc.↓	mIoU↑	F1↑	II↑	Unc.↓	mIoU↑	F1↑	[s]↓
Discrete	RL-Ours	**31.5** (0.12)	**38.3** (0.76)	**20.8** (0.26)	**25.6** (0.19)	30.5 (0.12)	**39.2** (0.42)	**20.4** (0.17)	**25.3** (0.12)	**0.004**
	RL-Base-D	31.1 (0.09)	38.6 (0.43)	20.7 (0.14)	25.5 (0.09)	30.4 (0.57)	40.2 (0.95)	20.1 (0.29)	25.0 (0.26)	**0.004**
	MCTS	30.7 (0.25)	41.9 (0.31)	19.5 (0.08)	24.5 (0.12)	**30.8** (0.21)	41.2 (0.85)	19.8 (0.31)	24.7 (0.24)	1.95
	CMA-ES	29.6 (1.87)	43.2 (2.38)	19.2 (0.85)	24.2 (0.76)	30.0 (1.45)	42.4 (0.61)	19.5 (0.42)	24.4 (0.51)	3.75
	Greedy	29.9 (0.24)	44.4 (0.83)	18.7 (0.26)	23.8 (0.22)	29.4 (0.17)	45.6 (0.59)	18.2 (0.22)	23.2 (0.22)	0.03
	Coverage	29.7 (0.46)	44.3 (0.37)	18.7 (0.12)	23.8 (0.12)	27.9 (0.24)	45.3 (0.37)	18.4 (0.12)	23.3 (0.08)	-



Feature field F | Map belief $\hat{F}_t$ | Unified belief $\hat{F}_{I,t}$

Figure 5.4: (Top-left) Simulated continuous-valued terrain feature field with values from low to high indicated as a colour gradient from blue to red. Shaded areas indicate non-interesting areas with low values in dark blue. (Bottom-left) Simulated discrete-valued terrain feature field with different colours indicating the semantic classes. The red class indicates areas of interest. (Top-centre) Mission-specific Gaussian process map belief $\hat{F}_t$. (Bottom-centre) Mission-specific occupancy map belief $\hat{F}_t$, where grey areas are unknown. (Right) Our fully probabilistic and map-agnostic unified belief over areas of interest $\hat{F}_{I,t}$. The probability of an area being of interest, ranging from low to high, is indicated as a colour gradient from blue to yellow.



Temperate-1 | Temperature-2 | Potsdam [68] | RIT-18 [75]

Figure 5.5: Real-world terrain orthomosaic datasets. (Left) Crop field surface temperatures, where high surface temperature (red) indicates areas of interest. (Right) Semantic urban and rural terrains, where vegetation (green and light blue) indicates areas of interest.

Noticeably, our map-agnostic policy outperforms the map-specific RL-Base-C and RL-Base-D policies on varying hyperparameters that cause larger inter-mission variations. This verifies the advantage of our unified policy being trained and conditioned on larger mission variations. In contrast, the RL-Base-C policy trained on static hyperparameters does not match the performance of the online policy search methods. This verifies that the learning-based adaptive IPP baselines degrade in performance as user-defined mission characteristics, such as interesting feature values and map hyperparameters, change between missions.

Furthermore, our map-agnostic policy outperforms the strongest MCTS adaptive IPP method on missions with varying hyperparameters while substantially reducing replanning runtimes at deployment. This shows that we can successfully train a single map-agnostic policy, which is applicable and well-performing in monitoring scenarios with large variations in user-defined hyperparameters and terrain map representations. Fig. 5.4 shows simulated ground truth feature fields F with paths planned based on our unified belief $\hat{F}_{I,t}$ over initially unknown non-shaded and red areas of interest ξ_I, derived from mission-specific map beliefs $\hat{F}_t$ with yellow indicating a high probability of interesting areas according to $\hat{F}_{I,t}$.

5.2.3 Results on Real-World Datasets

The experiments on real-world orthomosaics illustrated in Fig. 5.5 are designed to assess our second claim. Our results verify that our map-agnostic policy trained in simulation performs similarly to state-of-the-art adaptive IPP methods on previously unseen real-world terrain datasets. We compare our policy RL-Ours to map-specifically designed non-learning-based planning methods and map-specifically trained policies RL-Base-C/D. We consider two continuous-valued surface temperature orthomosaics of crop fields near Bonn, Germany, mapped using Gaussian processes, where high surface temperatures above 25°C are interesting (red). Furthermore, we execute discrete-valued semantic monitoring of an urban area in Potsdam, Germany [68] and a rural area [75] (RIT-18), mapped using occupancy maps, where vegetation features are of interest (green and light blue).

Table 5.4: Comparison of state-of-the-art map-specifically designed and trained methods to our map-agnostic planning policy (RL-Ours) on real-world continuous-valued surface temperature (Temperature-1/2) and discrete-valued urban (Potsdam) and rural (RIT-18) semantic terrain datasets. Best average performances are marked in bold, and second-best average performances are underlined if standard deviations in brackets overlap. Our map-agnostic policy performs similarly to state-of-the-art adaptive IPP methods.

Approach	**Temperature-1**		**Temperature-2**		**Potsdam** [68]		**RIT-18** [75]	
	II↑	Unc.↓	II↑	Unc.↓	II↑	Unc.↓	II↑	Unc.↓
RL-Ours	**25.4** (0.31)	62.2 (0.09)	27.6 (0.26)	58.6 (0.22)	**32.9** (1.92)	**35.8** (3.10)	31.7 (1.25)	39.9 (0.64)
RL-Base	24.6 (0.62)	63.1 (0.46)	26.7 (0.14)	60.6 (0.85)	31.9 (1.82)	36.4 (2.24)	**32.2** (0.31)	**39.6** (0.59)
MCTS	25.3 (0.22)	**61.9** (0.21)	**27.7** (0.49)	**58.4** (0.49)	31.7 (0.45)	40.8 (0.93)	30.7 (0.50)	42.2 (0.17)
Greedy	25.2 (0.46)	62.4 (0.45)	27.1 (0.43)	58.9 (0.29)	29.8 (0.29)	44.0 (0.66)	29.7 (0.45)	46.0 (0.54)
Coverage	14.7 (0.34)	75.9 (0.08)	15.8 (0.84)	73.4 (0.98)	29.3 (0.60)	44.6 (0.17)	29.7 (0.62)	45.4 (0.91)

Tab. 5.4 summarises the real-world dataset results. In line with Chap. 4 and state-of-the-art baselines, our map-agnostic policy consistently outperforms non-adaptive coverage paths, which ignore knowledge collected about the terrain during deployment. This showcases the advantages of adaptive online replanning for robotic information gathering compared to traditionally used pre-planned coverage paths. Notably, our map-agnostic policy outperforms greedy planning, highlighting the performance gain of non-myopically learned adaptive IPP policies on real-world datasets. Additionally, our policy performs similarly to the strong MCTS online policy search method while substantially reducing replanning runtimes from seconds to milliseconds as shown in Tab. 5.3. Furthermore, our map-agnostic policy performs better than the map-specifically designed and trained RL-Base-C and RL-Base-C policies on most real-world datasets. This highlights the strong performance of our policy in previously unseen terrains, which is potentially due to its training on diverse simulated monitoring missions.

While our map-agnostic policy is directly applied to all real-world terrain dataset missions without adaptation or re-training, greedy and MCTS methods require re-design for continuous- and discrete-valued terrain feature monitoring missions as explained in Sec. 5.2.1. The learning-based policies RL-Base-C and RL-Base-D even require re-training. Particularly, the RL-Base-D policy can only be applied to semantic monitoring missions with the same number of classes used for training, further complicating the deployment of map-specifically trained policies. These results highlight the advantages of our map-agnostic policy, validating its performance on real-world terrain data while facilitating deployment.

5.2.4 Map-Agnostic Online Planning Policy Search

The next set of experiments investigates our third claim. We demonstrate that our map-agnostic formulation integrates with and unifies state-of-the-art online

Table 5.5: Integration of our map-agnostic adaptive IPP formulation (ours) into state-of-the-art online policy search methods. Best average performances are marked in bold, and second-best average performances are underlined if standard deviations in brackets overlap. Our map-agnostic formulation maintains or improves performance over previous map-specific formulations for continuous- (prev-C) and discrete-valued (prev-D) terrain feature monitoring missions.

	Policy	**IPP**	**Varying $\mathcal{H}$**			
			II↑	Unc.↓	MLL↓	RMSE↓
Continuous	Greedy	prev-C	25.2 (0.66)	61.7 (0.29)	-58.0 (1.55)	4.35 (0.24)
		ours	**25.3** (0.41)	**61.3** (0.42)	**-59.0** (0.73)	**4.15** (0.16)
	MCTS	prev-C	25.3 (0.24)	61.1 (0.24)	-59.5 (0.70)	4.27 (0.22)
		ours	**27.0** (0.42)	**59.6** (0.26)	**-63.8** (0.34)	**4.00** (0.24)
	CMA-ES	prev-C	21.5 (1.44)	**64.3** (2.32)	**-55.4** (5.52)	**2.59** (0.85)
		ours	**21.8** (2.25)	64.5 (1.97)	-54.1 (5.67)	2.73 (1.26)
			II↑	Unc.↓	mIoU↑	F1↑
Discrete	Greedy	prev-D	29.4 (0.17)	45.6 (0.59)	18.2 (0.22)	23.2 (0.22)
		ours	**30.5** (0.33)	**44.5** (0.19)	**18.6** (0.00)	**23.6** (0.05)
	MCTS	prev-D	30.8 (0.21)	41.2 (0.85)	**19.8** (0.31)	**24.7** (0.24)
		ours	**31.4** (0.78)	**41.0** (0.96)	**19.8** (0.31)	**24.7** (0.31)
	CMA-ES	prev-D	**30.0** (1.45)	42.4 (0.61)	19.5 (0.42)	24.4 (0.51)
		ours	29.6 (1.33)	**41.6** (0.37)	**19.7** (0.21)	**24.6** (0.42)

non-learning-based policy search methods developed for adaptive IPP while maintaining or improving their performance in various terrain monitoring missions.

To demonstrate the universal applicability of our approach, we integrate our map-agnostic adaptive IPP formulation (ours) with the greedy, MCTS and CMA-ES policy search algorithms used in state-of-the-art non-learning-based adaptive IPP methods [30, 61, 117, 126]. We compare our formulation to previously used map-specific adaptive IPP formulations for continuous- (prev-C) and discrete-valued terrain feature monitoring (prev-D) described in Sec. 5.2.1.

Tab. 5.5 summarises the planning performances of the different online policy search methods. Our map-agnostic adaptive IPP formulation consistently performs on par with adaptive IPP formulations specifically designed for continuous- and discrete-valued monitoring missions, irrespective of the policy search method. Notably, our map-agnostic formulation even improves the average planning performance of policy search algorithms in some scenarios. These results verify that our method successfully integrates with state-of-the-art adaptive IPP methods without requiring map-specific adaptation. In this way, our approach contributes to unifying the broad family of adaptive IPP approaches.

5.3 Conclusion

Depending on the information-gathering mission, different environmental feature information is considered for mapping. Continuous-valued features, such as surface temperature, are mapped using Gaussian processes or Kalman filters, while discrete-valued features, such as semantic segmentation of urban areas, are mapped using occupancy maps. Previous adaptive IPP methods consider either continuous- or discrete-valued feature monitoring missions. They directly integrate the respective environment map as their state representation used for planning. Thus, these approaches require re-designing the state representation as the to-be-mapped environment features change. Learning-based adaptive IPP policies are trained on these mission-specific state representations, hence even requiring re-training as to-be-mapped environment features change.

To address this issue, we have proposed a novel map-agnostic formulation of the adaptive IPP problem in this chapter. Our adaptive IPP formulation is generally applicable to various continuous- or discrete-valued terrain feature monitoring missions. Our main contribution is a planning state representation unifying different map representations. Based on our planning state representation and a newly introduced reward function for adaptive IPP, we train a single planning policy with RL on terrain monitoring missions with varying map representations and user-defined areas of interest. Our experimental results show that our learned map-agnostic policy yields competitive performance with state-of-the-art online non-learning-based policy search methods and offline-learned policies re-designed and re-trained for each class of monitoring missions with specific map representations. Moreover, our planning policy trained in simulation performs similarly to state-of-the-art adaptive IPP methods on unseen real-world datasets. Furthermore, our map-agnostic adaptive IPP formulation unifies existing adaptive IPP methods while maintaining or improving planning performance.

Our experimental results demonstrate that, in response to the first research question posed in this thesis, modelling the adaptive IPP problem in a map-agnostic fashion opens up the path for RL-based adaptive IPP methods that increase the compute efficiency while providing competitive information-gathering performance for a large family of terrain monitoring missions. However, in monitoring missions during which the robot is tasked to map discrete-valued semantics, such as monitoring buildings, streets, cars, and vegetation in cities for urban planning purposes, deep learning-based models perform semantic segmentation of collected images. These semantic segmentation models are commonly trained on costly static human-labelled datasets. As robots are deployed in unknown and changing environments, the collected images often deviate from the ones the semantic vision model was trained on. This often results in degraded semantic

segmentation prediction quality and, thus, in degraded performance in robotic information gathering. To this end, in Chap. 6, we propose a novel adaptive IPP method that aims to improve the robot's semantic vision in unknown environments while minimising the number of human-labelled images.

Chapter 6

Adaptive Informative Path Planning for Active Learning in Semantic Mapping

Adaptive informative path planning methods show encouraging performance in robotic information gathering as also shown in Chap. 4 and Chap. 5. They improve upon classically used non-adaptive preprogrammed information-gathering missions while ensuring a greater level of autonomous decision-making capabilities in unknown environments. However, these adaptive IPP methods, including the ones presented in Chap. 4 and Chap. 5, assume reliable onboard robotic vision to interpret collected sensor measurements semantically. Common monitoring applications that require the robot to have a pixel-wise semantic understanding of images are urban planning, inspection of industrial parks and monitoring of arable fields or rural areas.

Recent breakthroughs in computer vision [59, 81, 168] have enabled automated scene understanding in large-scale complex aerial environments [49, 75, 176] using deep learning-based vision models. These models are usually trained on a static curated human-labelled dataset. As the robot is deployed in unknown environments, sensor measurements often deviate from the ones the model was trained on. The visual appearance of environments might change, e.g. between two cities or as seasons change. Furthermore, the model may be pre-trained on another domain, e.g. urban images, and deployed at an industrial site. In both scenarios, the prediction quality of semantic segmentation models often degrades. This results in an overall degraded information-gathering efficiency. Thus, a critical requirement for robot autonomy and information gathering is the robot's ability to improve its deep learning-based vision with minimal expert guidance.

In this chapter, we examine the problem of active learning in UAV-based semantic mapping missions. Our goal is to improve the robot's semantic vision

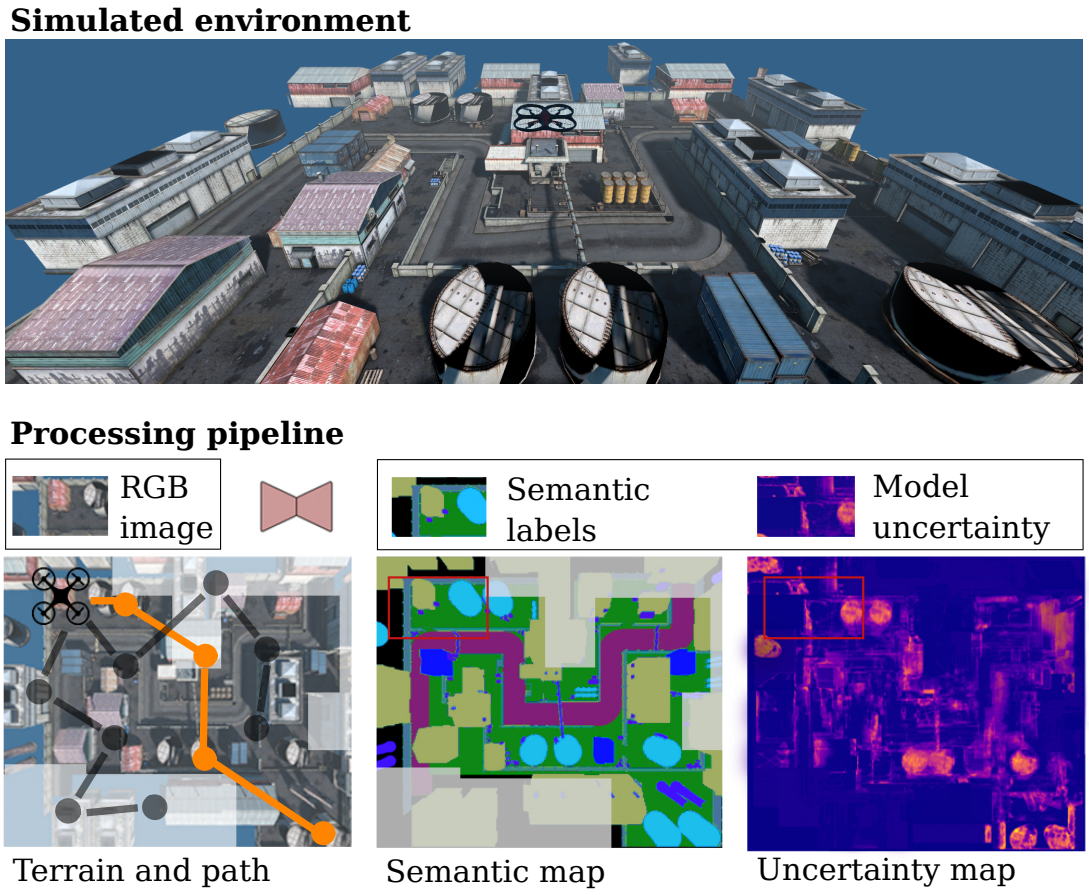


Figure 6.1: Our adaptive planning for active learning in UAV-based semantic mapping deployed in a photo-realistic simulator [157] (top). We compute an acquisition function, e.g. model uncertainty, and predict semantics (centre-right). We fuse both in terrain maps (bottom-right). Our map-based planners replan a UAV's path (orange, bottom-left) to collect the most informative, e.g. most uncertain (yellow, bottom-right), images for network re-training.

capabilities in unknown environments while aiming to minimise the number of human-labelled images required for training the vision model. To this end, we introduce a planning approach that exploits active learning ideas for computer vision. We incorporate active learning techniques into a new adaptive IPP framework for improving robotic semantic vision. The UAV replans its path online as new observations are collected to adaptively target areas of potentially informative new training data. The newly gathered images are semantically labelled by a human annotator. We re-train a semantic segmentation model based on the labelled data to maximise its prediction performance.

Various active learning methods reduce the requirements for human-labelled training data [17, 44, 71, 92, 93, 165, 173]. Recently, active learning approaches for deep learning models are gaining attention [40, 47, 147, 155, 183, 190]. These works develop acquisition functions for selecting to-be-labelled training data to maximise model performance. They assume access to large pre-recorded unla-

belled in-domain data pools to select training data from. Instead, we deploy the robot in unknown environments. Thus, we do not have access to a data pool before the robot's deployment. To avoid this issue, active learning works for aerial imagery use the UAV for passive data collection to record static data pools [74, 90]. In contrast, we address the thesis' second research question of how to improve robotic vision in unknown environments with a minimal number of human-labelled images by combining active learning and adaptive planning.

The main contribution of this chapter is a generally applicable adaptive IPP framework performing active learning of robotic semantic vision. Our framework's benefit is that it reduces the human labelling effort to improve robotic semantic vision in new and unseen terrains. A key novelty is to enable adaptive robotic data collection by linking ideas from active learning to planning objectives as illustrated in Fig. 6.1. To this end, we integrate various model uncertainty and training data novelty estimation techniques from active learning research [46, 84, 128] into deep learning-based semantic segmentation models [131, 132]. The inferred pixel-wise semantic labels, estimated model uncertainty and novelty scores are fused sequentially into a probabilistic terrain map as new observations are acquired. As a key feature, our framework iteratively replans the UAV's path to collect the most informative, i.e. the most uncertain or novel, images for human labelling and model re-training in a targeted fashion.

In sum, we make the following claims. First, our adaptive planning method reduces the number of human-labelled images required to maximise semantic segmentation performance compared to pre-programmed data collection campaigns and state-of-the-art local planning for active learning [12]. Second, our probabilistic global mapping of gathered information enhances map-based planning performance for active learning. Third, our Bayesian extension of the deterministic ERFNet [131] improves semantic segmentation performance and yields more consistent model uncertainty estimates, resulting in higher planning performance for active learning than previously used non-Bayesian planning objectives.

This chapter incorporates material from the following peer-reviewed journal and conference publications, for which I am the main contributor:

- Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. An Informative Path Planning Framework for Active Learning in UAV-Based Semantic Mapping. *IEEE Trans. on Robotics (TRO)*, 39(6):4279–4296, 2023

- Julius Rückin, Liren Jin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Informative Path Planning for Active Learning in Aerial Semantic Mapping. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022

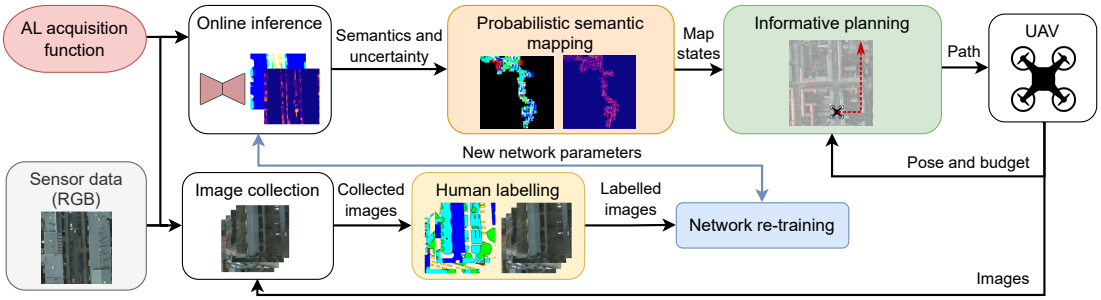


Figure 6.2: A pre-trained semantic segmentation network is deployed on a UAV. During a mission, the network predicts pixel-wise semantics, model uncertainties as outlined in Sec. 6.1.1.1, and novelty scores from RGB images as described in Sec. 6.1.1.2. This information is projected onto the terrain to build global maps capturing these variables as explained in Sec. 6.1.2. Based on the current UAV position, budget, and posterior map state, our algorithm adaptively replans paths for the UAV to collect potentially informative training data for improving the network performance as presented in Sec. 6.1.3. After the mission, the collected images are labelled by a human annotator and used for network re-training before re-deployment of the UAV.

6.1 Adaptive Informative Path Planning for Active Learning

The goal of our approach is to autonomously collect informative training data to improve the robot's semantic vision with a minimal number of human-labelled images. To achieve this, we present our adaptive IPP framework for active learning in UAV-based semantic mapping shown in Fig. 6.2. Our robot setup considers a UAV collecting images of a terrain using a downwards-facing RGB camera. Our framework links active learning with planning objectives guiding the UAV to areas of informative training data. As new data is collected, we use a lightweight fully convolutional neural network to predict pixel-wise semantics as described in Sec. 6.1.1. Moreover, we estimate the pixel-wise model uncertainty associated with the network's prediction as explained in Sec. 6.1.1.1 and training data novelty of the collected image as outlined in Sec. 6.1.1.2. Next, we fuse semantics, uncertainty estimates, and novelty scores into a probabilistic terrain map as detailed in Sec. 6.1.2. The UAV position, its remaining budget, and the current map state are combined into new active learning-based planning objectives used to adaptively replan the future path towards informative training data as presented in Sec. 6.1.3. A key feature of our framework is its general applicability. By design, it is agnostic to the chosen network architecture. It supports different uncertainty estimation techniques, mapping methods, and map-based planning strategies. The following subsections detail the framework's individual modules and the specific methods we investigate in this chapter.

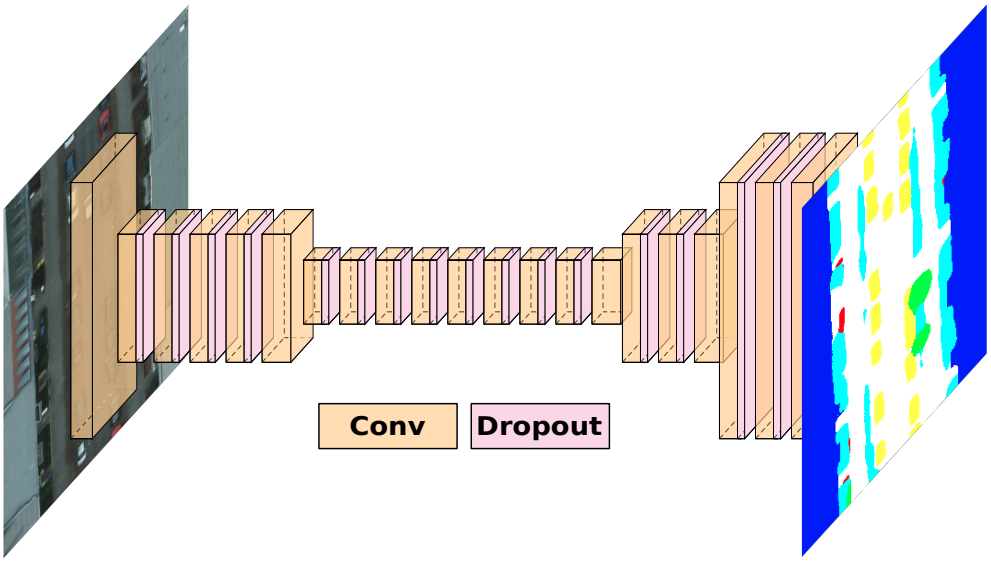


Figure 6.3: ERFNet architecture proposed by Romera et al. [131]. The network takes an RGB image (left) as input and outputs semantic labels (right). We use the network in our ensemble method to predict model uncertainty or by using dropout at train and inference time as described in Sec. 6.1.1.1. To this end, we add a dropout layer after each convolutional layer.

6.1.1 Active Learning Acquisition Functions

We first derive measures for an image's potential information value when a network is re-trained on this labelled data. To this end, active learning methods propose two main paradigms: uncertainty-based and representation-based acquisition functions. We demonstrate our framework using either paradigm.

We adapt the ERFNet encoder-decoder architecture proposed by Romera et al. [131] depicted in Fig. 6.3 to our active learning use case. We add a dropout layer after each convolutional layer to enable estimating model uncertainty. Although our framework is agnostic to the chosen network architecture, the lightweight ERFNet is particularly suitable for online robot deployment with limited computational resources. In the following, the model $f^{\mathbf{w}}(\cdot)$ is parameterised by weights $\mathbf{w} \in \mathbb{R}^D$ and outputs a probability tensor $p(\mathbf{y} \mid \mathbf{f}^{\mathbf{w}}(\mathbf{z})) = \text{softmax}(\mathbf{f}^{\mathbf{w}}(\mathbf{z})) \in [0,1]^{K \times w \times h}$, where $\mathbf{z} \in \{0, \dots, 255\}^{w \times h \times 3}$ is the RGB image with width w and height h, and $\mathbf{y} \in \{1, \dots, K\}^{w \times h}$ is the pixel-wise semantic label over K classes. The training set contains N images $\mathcal{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_N\}$ and semantic labels $\mathcal{Y} = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$. Our network is trained to minimise the cross-entropy loss function with weight decay factor λ,

$$\mathcal{L}(\mathbf{w}) = -\frac{1}{N} \sum_{i=1}^{N} \log \left(p\big(\mathbf{y}_i \mid \mathbf{f}^{\mathbf{w}}(\mathbf{z}_i)\big) \right) + \lambda \|\mathbf{w}\|_2^2 \,. \tag{6.1}$$

Sec. 6.1.1.1 and Sec. 6.1.1.2 describe uncertainty- and representation-based methods to estimate the information value of an image for network training.

6.1.1.1 Bayesian Uncertainty-Based Methods

We estimate pixel-wise model uncertainty over the prediction $p(\mathbf{y} \mid \mathbf{f}^{\mathbf{w}}(\mathbf{z}))$ as a measure for the informativeness of image $\mathbf{z}$ for re-training [8, 46, 47, 65, 76]. For this, we leverage advances in Bayesian deep learning, transforming the deterministic ERFNet into a probabilistic version. We consider using two alternative methods: Monte Carlo dropout [46] and ensembles [8]. To measure model uncertainty, we use Bayesian active learning by disagreement [65], which computes the mutual information between the unknown labels $\mathbf{y}$ and the posterior distribution over model weights $p(\mathbf{w} \mid \mathcal{Z}, \mathcal{Y})$. However, the posterior over the weights is intractable for neural networks due to their high-dimensional parameter space $\mathbf{w} \in \mathbb{R}^D$ [46]. Thus, we approximate the true posterior prediction [76] by

$$\hat{p}(\mathbf{y} \mid \mathbf{z}, \mathcal{Z}, \mathcal{Y}) = \frac{1}{T} \sum_{i=1}^{T} \operatorname{softmax}\big(\mathbf{f}^{\hat{\mathbf{w}}_i}(\mathbf{z})\big) \,, \tag{6.2}$$

where we independently sample T weights $\hat{\mathbf{w}}_i \sim q(\mathbf{w})$ from a prior distribution $q(\mathbf{w})$ performing Monte Carlo integration over model weights $\mathbf{w}$.

Monte Carlo dropout and ensemble methods provide two alternative approaches to construct the prior $q(\mathbf{w})$. In Monte Carlo dropout, dropout is applied independently to the weights $\mathbf{w}$ before each of the T forward passes at test time. In the ensemble method, we train T independently randomly initialised models $f^{\hat{\mathbf{w}}_i}$ with stochastic mini-batch gradient descent. For further details on Monte Carlo dropout and ensembles, we refer to Gal and Ghahramani [46] and Beluch et al. [8], respectively. Following Gal et al. [47], we approximate the mutual information as a measure of model uncertainty using Eq. (6.2)

$$\begin{aligned} \mathbb{I}(\mathbf{y}, \mathbf{w} \mid \mathbf{z}, \mathcal{Z}, \mathcal{Y}) \approx & -\hat{p}(\mathbf{y} \mid \mathbf{z}, \mathcal{Z}, \mathcal{Y})^{\top} \log\big(\hat{p}(\mathbf{y} \mid \mathbf{z}, \mathcal{Z}, \mathcal{Y})\big) \\ & + \frac{1}{T} \sum_{i=1}^{T} p(\mathbf{y} \mid \mathbf{z}, \hat{\mathbf{w}}_i)^{\top} \log\big(p(\mathbf{y} \mid \mathbf{z}, \hat{\mathbf{w}}_i)\big) \,, \end{aligned} \tag{6.3}$$

where $\log(\cdot)$ is applied element-wise. Intuitively, this mutual information is high whenever the average prediction entropy is high (first term) while single predictions' entropies are low (second term). Thus, predictions are certain but disagree with each other. We exploit this measure to guide the UAV towards more informative areas, i.e. areas of high model uncertainty. Note that our framework is agnostic to both the model uncertainty estimation method and the network.

6.1.1.2 Representation-Based Method

Inspired by recent active learning works [40, 147, 155], we study a representation-based planning objective as an alternative to uncertainty-based objectives. We deterministically quantify the network's prediction confidence by estimating the

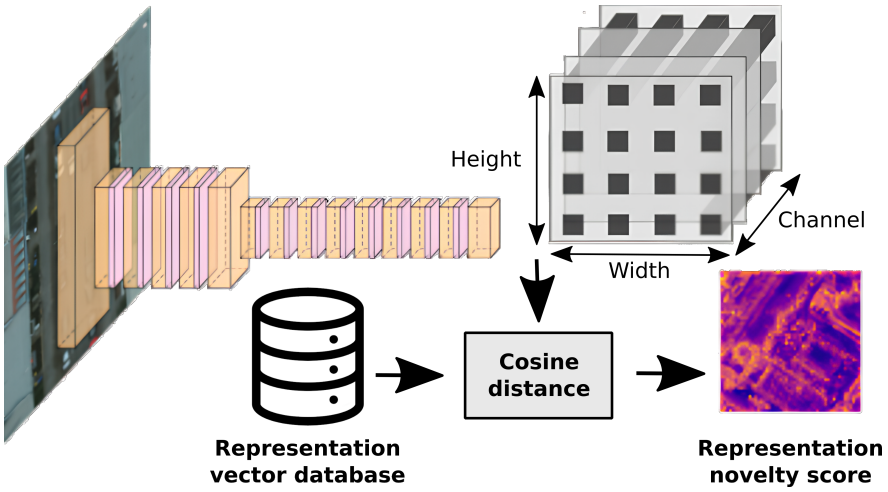


Figure 6.4: Representation-based image novelty score [12]. An RGB image $\mathbf{z}$ is passed through our ERFNet encoder. Its latent vectors $\mathbf{r}^{\mathbf{z}}_{i,j}$ are extracted along the channel dimension. We compute the cosine distance between each $\mathbf{r}^{\mathbf{z}}_{i,j}$ and its k-nearest neighbours from the representation vector database consisting of all training images' $\mathcal{Z}$ latent vectors. The resulting novelty image is upsampled to the spatial dimensions of $\mathbf{z}$. Last, we add all $\mathbf{r}^{\mathbf{z}}_{i,j}$ to the representation vector database. Lighter colours indicate higher novelty, i.e. higher informativeness for re-training.

image's novelty to the network $f^{\mathbf{w}}$ given training images and labels $\mathcal{Z}, \mathcal{Y}$ similar to prior work [12, 101, 121, 128]. Intuitively, the image's novelty is high whenever the network's latent representations of a new image $\mathbf{z}$ and training images $\mathcal{Z}$ are dissimilar. Although confidence measures for classification are well-known [101, 121], they are not directly applicable to semantic segmentation as they do not provide pixel-wise scores and are not invariant to object locations. Thus, we use the novelty measure for semantic segmentation proposed by Blum et al. [12].

We perform kernel-density estimation in the network's latent space by computing the average cosine distance between the latent representations of image $\mathbf{z}$ and its k-nearest latent representations of training images $\mathcal{Z}$. We exploit the fully convolutional neural network's architecture, where the network $f^{\mathbf{w}}(\cdot) = d^{\mathbf{w}_d}(e^{\mathbf{w}_e}(\cdot))$ consists of an encoder $e^{\mathbf{w}_e}$ parameterised by $\mathbf{w}_e \in \mathbb{R}^{D_e}$, a decoder $d^{\mathbf{w}_d}$ parameterised by $\mathbf{w}_d \in \mathbb{R}^{D_d}$. Specifically, we extract representations $\mathbf{e}^{\mathbf{w}_e}(\mathbf{z}) = \mathbf{r}^{\mathbf{z}} \in \mathbb{R}^{\frac{w}{8} \times \frac{h}{8} \times C}$ after the encoder's last convolutional layer, where C are the channel dimensions. Induced by the ERFNet architecture, spatial dimensions are downsampled by a factor of 8 compared to the image. Hence, $\mathbf{r}^{\mathbf{z}}_{i,j} \in \mathbb{R}^C$ is a C-dimensional latent vector of the (i,j)-th 8×8 pixels patch of image $\mathbf{z}$. After model training, we generate a database $R = \left\{ \mathbf{r}^{\mathbf{z}_1}_{1,1}, ..., \mathbf{r}^{\mathbf{z}_N}_{\frac{w}{8},\frac{h}{8}} \right\}$ of $\frac{w}{8} \cdot \frac{h}{8} \cdot N$ patch-wise representations of the training images $\mathcal{Z}$. Given an image $\mathbf{z}$ at inference time, its (i,j)-th novelty score is computed as

$$r(\mathbf{z})_{i,j} = \frac{1}{k} \sum_{\mathbf{r} \in \mathrm{NN}(\mathbf{r}^{\mathbf{z}}_{i,j})} 1 - \left| \frac{\mathbf{r}^\top \mathbf{r}^{\mathbf{z}}_{i,j}}{\|\mathbf{r}\|_2 \, \|\mathbf{r}^{\mathbf{z}}_{i,j}\|_2} \right|, \tag{6.4}$$

where $\mathrm{NN}(\mathbf{r}^{\mathbf{z}}_{i,j})$ is the set of k-nearest neighbors of $\mathbf{r}^{\mathbf{z}}_{i,j}$ in R using the cosine distance as the distance metric. Intuitively, higher novelty indicates higher informativeness of image $\mathbf{z}$ for re-training. Fig. 6.4 provides a schematic of an image's novelty score computation. For more details, we refer to Blum et al. [12].

Our framework can easily be adapted to other acquisition functions and fully convolutional neural networks. In our experimental evaluation in Sec. 6.2, we showcase our framework's generality using the uncertainty- and representation-based objectives with ERFNet [131] and U-Net [132], as described above.

6.1.2 Probabilistic Semantic Mapping

An important basis for our new planning objective functions is our 2D multi-layer terrain map. This map captures global semantics, training data statistics, as well as model uncertainties and representation novelties introduced in Sec. 6.1.1. In this way, the map provides different sources of information for adaptive replanning of paths. We probabilistically update the map information online as the UAV collects new images. To achieve this, we use sequential occupancy grid mapping [114] to update each map layer when a new measurement arrives. We discretise the terrain into three 2D maps $\mathcal{G}_S : G \to \{0,1\}^{K\times W\times L}$, $\mathcal{G}_U : G \to [0,1]^{W\times L}$, and $\mathcal{G}_R : G \to [0,1]^{W\times L}$ defined over a grid lattice G with $W \times L$ spatially independent cells. These maps spatially capture the terrain's semantic classes $\mathcal{G}_S$, model uncertainties $\mathcal{G}_U$, and training data novelty scores $\mathcal{G}_R$.

The semantic map $\mathcal{G}_S$ consists of K independent layers $\mathcal{G}_{S_i} : G \to \{0,1\}^{W\times L}$ to map all K classes $i \in \{1,\dots,K\}$. Each grid cell's c initial state follows a uniform prior distribution $\mathcal{G}_{S_i}(c) \sim p(\mathcal{G}_{S_i}(c) = 1) = \frac{1}{K}$. When a new image $\mathbf{z}_t$ arrives at time step t, the semantic predictions $\hat{p}(\mathbf{y} \mid \mathbf{f}^{\mathbf{w}}(\mathbf{z}_t))$, see Eq. (6.2), are projected to the flat terrain given the UAV position $\mathbf{p}_t \in \mathbb{R}^3$ and camera intrinsics. We use standard occupancy grid mapping [114] for each layer i and cell c computing the posterior belief $\mathcal{G}_{S_i}(c) \sim p(\cdot \mid \mathbf{z}_{1:t}, \mathbf{p}_{1:t})$ as

$$l(\mathcal{G}_{S_i}(c) \mid \mathbf{z}_{1:t}, \mathbf{p}_{1:t}) = l(\mathcal{G}_{S_i}(c) \mid \mathbf{z}_t, \mathbf{p}_t) + l(\mathcal{G}_{S_i}(c) \mid \mathbf{z}_{1:t-1}, \mathbf{p}_{1:t-1}) - l(\mathcal{G}_{S_i}(c))\,, \quad (6.5)$$

where $l(\cdot)$ are the log odds of the binary random variable, $p(\mathcal{G}_{S_i}(c) \mid \mathbf{z}_t, \mathbf{p}_t)$ is given by the projected semantic predictions, $p(\mathcal{G}_{S_i}(c) \mid \mathbf{z}_{1:t-1}, \mathbf{p}_{1:t-1})$ is the recursive map belief, and $p(\mathcal{G}_{S_i}(c))$ is the map prior.

The model uncertainties and novelties are stored in the maps $\mathcal{G}_U$ and $\mathcal{G}_R$ with prior means $\mu_{U,0}$ and $\mu_{R,0}$ respectively. We fuse projected uncertainties $\mathbf{u}_t$ given by Eq. (6.3) and novelty scores $r(\mathbf{z}_t)$ given by Eq. (6.4) using maximum likelihood estimation assuming normally distributed $\mathcal{G}_U$ and $\mathcal{G}_R$. To this end, we maintain a hit map $H : G \to \mathbb{N}^{W\times L}$, counting the total number of times a grid cell was

updated. Then, we update the means $\mu_{U,t}(c)$ and $\mu_{R,t}(c)$ for a grid cell $c \in G$ by

$$\mu_{U,t}(c) = \mu_{U,t-1}(c) + \frac{1}{H(c)} \left(\mathbf{u}_t^c - \mu_{U,t-1}(c)\right) , \tag{6.6}$$

$$\mu_{R,t}(c) = \mu_{R,t-1}(c) + \frac{1}{H(c)} \left(r(\mathbf{z}_t)^c - \mu_{R,t-1}(c)\right) , \tag{6.7}$$

where $\mathbf{u}_t^c$ and $r(\mathbf{z}_t)^c$ are uncertainties and novelty scores computed as in Sec. 6.1.1 and projected to grid cell c on the flat terrain. Last, a map $T : G \to \mathbb{N}^{W \times L}$ counts how often grid cells occur in the training data set to foster data diversity in our proposed planning objectives. Note that the maps $H(\cdot)$ and $T(\cdot)$ are different as the camera could provide a high-frequency image stream for mapping while images only at the planned measurement position are collected for training.

A key feature of our mapping approach is that we accumulate and update the information between missions by updating the map prior. After each UAV mission, the network is re-trained on the collected training data. Re-training changes the semantic predictions, model uncertainty, and representation novelty estimates. Thus, we store all previously collected images and corresponding UAV positions. After re-training, we predict semantics, model uncertainties, and representation novelties of the stored images and sequentially fuse them into the maps. Our informed map prior strategy enhances map-based planning by avoiding exploring from scratch or replanning with outdated terrain knowledge.

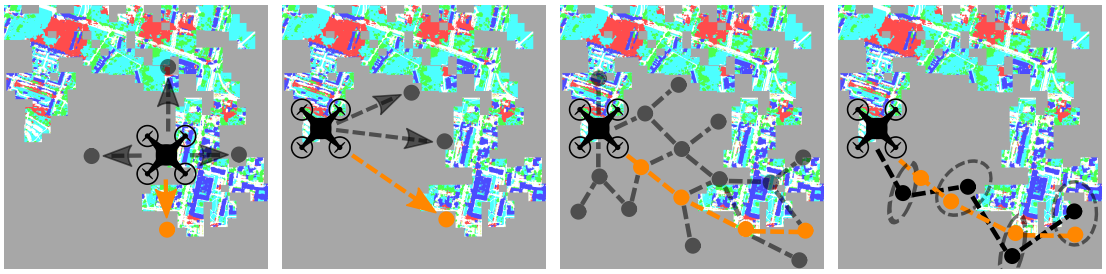
6.1.3 Adaptive Informative Path Planning Algorithms

We develop adaptive IPP algorithms to guide a UAV to adaptively collect useful training data for our fully convolutional neural network. Our key idea is to link acquisition functions introduced in Sec. 6.1.1 to planning objective functions. Our planning strategies use the probabilistic terrain maps presented in Sec. 6.1.2 to guide the UAV online towards informative training data in an unknown terrain.

Adaptive IPP algorithms optimise an information criterion $I : \Psi \to \mathbb{R}_{\geq 0}$ over paths $\psi = (\mathbf{p}_1, \ldots, \mathbf{p}_P) \in \Psi$ defined by P waypoints $\mathbf{p}_i \in \mathbb{R}^3$ considering a mission budget $B \geq 0$ and cost function C, where Ψ is the set of all possible paths of length P. The function $C : \Psi \to \mathbb{R}_{\geq 0}$ defines the cost of executing a path ψ as

$$C(\psi) = \sum_{i=1}^{P-1} d(\mathbf{p}_i, \mathbf{p}_{i+1}) , \tag{6.8}$$

where $d : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}_{\geq 0}$ computes the travel time between two waypoints assuming constant acceleration and deceleration $\pm a$, and maximum velocity v. The key insight of our method is to couple the active learning acquisition functions with IPP information criteria I. This allows us to maximise model performance and minimise the number of human-labelled images collected along path ψ.



(a) Local (b) Frontier (c) Sampling (d) Optimisation

Figure 6.5: Our planners for training data collection. Dark gray dots and lines indicate candidate measurements and paths evaluated based on their estimated information adding these images to the training set. Orange dots and lines indicate the most informative chosen measurements and paths. Light grey depicts unexplored terrain. In (d), black indicates the greedy initialisation, and ellipses indicate the optimisation of candidate paths in continuous space.

We propose four different replanning strategies in our framework: one local image-based and three global frontier, optimisation and sampling schemes that optimise information criteria I given the current terrain map state. The planners are illustrated in Fig. 6.5. In the following, we exemplarily present our planning objectives with respect to the globally mapped model uncertainties $\mathcal{G}_{U,t}$ at a time step t, see Eq. (6.6). In case of the representation-based objective, we substitute the uncertainties $\mathcal{G}_{U,t}$ with novelties $\mathcal{G}_{R,t}$, see Eq. (6.7). Our mapping and planning methods are agnostic to the active learning acquisition function computed for individual images as in Sec. 6.1.1. Thus, one can extend our framework with other acquisition functions, as we showcase in our experimental evaluation.

Local planner. Our local planner follows the direction of the highest estimated training data information in the image $\mathbf{z}_t$ recorded at the UAV position $\mathbf{p}_t$. We select the image edge $e^*_{\mathbf{z}_t}$ with the highest image-based active learning value, e.g. model uncertainties $\mathbf{u}_t$, normalised by the respective training data counts in $T_t(\mathbf{p}_t)$. Training data counts $T_t(\mathbf{p}_t)$ are extracted from the map by projecting the camera's field of view from position $\mathbf{p}_t$ on the flat terrain. Like this, we select neighbouring informative images while locally fostering training data diversity. The next-best measurement position $\mathbf{p}^*_{t+1}$ is then reached by taking a predefined step size in the direction of edge $e^*_{\mathbf{z}_t}$ at a fixed altitude. This resembles the planner proposed by Blum et al. [12] and generalises it to any active learning objective.

Frontier-based planner. Our global geometric planner guides the UAV towards the frontiers of the explored terrain [182] with the highest active learning objective in the terrain map. We use the hit map H to identify exploration frontiers. A grid cell $c_k \in G$ is considered to be known if its hit count is $H(c_k) > 0$. A grid cell $c_u \in G$ is considered to be unknown if its hit count is $H(c_u) = 0$. Frontiers are defined as a connected set of known grid cells c_u with neighbouring unknown grid cells. We sample a set of next candidate measurement positions $\mathcal{P}^c_{t+1}$

equidistantly along the frontiers at a fixed altitude. Then, we greedily choose the next-best measurement position $\mathbf{p}^*_{t+1} \in \mathcal{P}^c_{t+1}$ from the set of next candidate positions. Since the frontier-based planner acts greedily, optimising the information criterion I reduces to selecting the next-best path $\psi^* = (\mathbf{p}^*_{t+1})$ with

$$\mathbf{p}^*_{t+1} = \underset{\mathbf{p}^c_{t+1} \in \mathcal{P}^c_{t+1}}{\operatorname{argmax}} \; I((\mathbf{p}^c_{t+1})) = \underset{\mathbf{p}^c_{t+1} \in \mathcal{P}^c_{t+1}}{\operatorname{argmax}} \; \frac{\|\mathcal{G}_{U,t}(\mathbf{p}^c_{t+1})\|_1}{\|T_t(\mathbf{p}^c_{t+1})\|_1}\,, \tag{6.9}$$

where $\mathcal{G}_{U,t}(\mathbf{p}^c_{t+1})$ and $T_t(\mathbf{p}^c_{t+1})$ are the currently globally mapped model uncertainties and training data counts within the camera's field of view from position $\mathbf{p}^c_{t+1}$, and $\|\cdot\|_1$ is the vector norm summing all elements. In this way, our frontier planner balances the exploration of unknown space for training data diversity and focuses on areas potentially valuable for maximising model improvement.

Optimisation-based planner. Our optimisation-based planner selects a path ψ^*_{t+1} over a fixed horizon of multiple time steps. We use a two-step approach for efficient online replanning inspired by Popović et al. [126]. As a first step, we greedily select a path ψ^g_{t+1} of length P over a grid above the terrain. In the second step, we use an optimisation procedure to fine-tune ψ^g_{t+1} in the continuous UAV workspace and return the next-best path ψ^*_{t+1}.

We iteratively select a path $\psi^g_{t+1} = (\mathbf{p}^g_{t+1}, \ldots, \mathbf{p}^g_{t+P})$, where each measurement position $\mathbf{p}^g_{t+i}$, $i \in \{1, \ldots, P\}$ is chosen over a sparse lattice $\mathcal{F}$ of candidate positions $\mathbf{p}^c \in \mathcal{F}$ at a fixed altitude. We sequentially and greedily choose measurement positions $\mathbf{p}^g_{t+i}$ for $i \leq P$ time steps into the future as

$$\mathbf{p}^g_{t+i} = \underset{\mathbf{p}^c \in \mathcal{F}}{\operatorname{argmax}} \; \frac{\|\mathcal{G}_{U,t}(\mathbf{p}^c)\|_1}{d(\mathbf{p}_{t+i-1}, \mathbf{p}^c)\|T_{t+i-1}(\mathbf{p}^c)\|_1}\,, \tag{6.10}$$

where $T_{t+i-1}(\mathbf{p}^c)$ is the subset of the forward-simulated training data count map given by the camera's field of view at position $\mathbf{p}^c$. The forward simulation of the current map T_t is computed based on the previously selected positions $(\mathbf{p}^g_{t+1}, \ldots, \mathbf{p}^g_{t+i-1})$ and their respective camera's field of views. Forward-simulating T_t is crucial as one cannot forward-simulate model uncertainties $\mathcal{G}_{U,t}$ during planning. Forward-simulating T_t linearly decreases uncertainty with the number of training set occurrences. This fosters data diversity and terrain exploration.

Subsequently, we refine the greedy positions of ψ^g_{t+1} in parallel in the continuous UAV workspace. To this end, we initialise an optimisation procedure with the greedy solution ψ^g_{t+1} and extend Eq. (6.10) to an information criterion I evaluating candidate paths $\psi^o_{t+1} = (\mathbf{p}^o_{t+1}, \ldots, \mathbf{p}^o_{t+P})$ by

$$I(\psi^o_{t+1}) = \sum_{i=1}^{P} \frac{\|\mathcal{G}_{U,t}(\mathbf{p}^o_{t+i})\|_1}{d(\mathbf{p}^o_{t+i-1}, \mathbf{p}^o_{t+i})\|T_{t+i-1}(\mathbf{p}^o_{t+i})\|_1}. \tag{6.11}$$

The candidate path $\psi^*_{t+1} = (\mathbf{p}^*_{t+1}, \ldots, \mathbf{p}^*_{t+P})$ maximising Eq. (6.11) is chosen and measurement position $\mathbf{p}^*_{t+1}$ is executed. We found that normalising active

learning information of a path by its execution costs $d(\cdot)$ leads to more efficient budget allocation. This planner supports any optimisation algorithm, which can optimise Eq. (6.11), e.g. derivate-free evolutionary algorithms [57].

Sampling-based planner. Our sampling-based planner uses Monte Carlo tree search [13] to optimise a next-best measurement position $\mathbf{p}_{t+1}^*$ in a non-myopic fashion. We simulate a number of future paths $\psi_{t+1} = (\mathbf{p}_{t+1}^{n_1}, \dots, \mathbf{p}_{t+P}^{n_P})$ of length P at a fixed altitude. Each node n_i at depth $i \in \{0, \dots, P\}$ in the tree encodes a possible measurement position $\mathbf{p}_{t+i}$ in a simulated path ψ_{t+1}. Thus, a node n_i is uniquely defined by its state $S^{n_i} = \left\{\mathbf{p}_{t+i}^{n_i}, T_{t+i}^{n_i}, B_{t+i}^{n_i}\right\}$ consisting of a measurement position $\mathbf{p}_{t+i}^{n_i}$, forward-simulated training data count map $T_{t+i}^{n_i}$ along the traversed path in the tree to node n_i, and remaining budget $B_{t+i}^{n_i}$. The tree's root node n_0 is defined by $S^{n_0} = \{\mathbf{p}_t, T_t, B_t\}$, where $\mathbf{p}_t$, T_t and B_t are the current UAV position, training data count map, and remaining budget. At each node, the planner selects the next position from a discrete set of actions with different step sizes and orientations. While traversing the search tree, we use the upper confidence bound bandit algorithm [13] to choose a child node. When reaching a leaf node n_i, we roll out the remaining $P - i$ measurement positions along the path by sampling actions uniformly at random until the remaining budget is exceeded or path length P is reached. A simulated path's information value $I(\psi_{t+1}) = \sum_{i=0}^{P-1} R(n_i, n_{i+1})$ is computed by summing rewards $R(n_i, n_{i+1})$ along subsequent parent and child nodes n_i, n_{i+1} given by

$$R(n_i, n_{i+1}) = \frac{\|\mathcal{G}_{U,t}(\mathbf{p}_{t+i+1}^{n_{i+1}})\|_1}{d(\mathbf{p}_{t+i}^{n_i}, \mathbf{p}_{t+i+1}^{n_{i+1}})\|T_{t+i}^{n_i}(\mathbf{p}_{t+i+1}^{n_{i+1}})\|_1}. \tag{6.12}$$

Note that $T_{t+i}^{n_i}(\mathbf{p}_{t+i+1}^{n_{i+1}})$ are the training data occurrences at the child node's position $\mathbf{p}_{t+i+1}^{n_{i+1}}$ after forward-simulating the current training data count T_t from the root node n_0 along the traversed tree path based on the carema's field of views up to the parent's node position. Like this, the reward function estimates the next position's information value given the training data count at replanning time $t+i$. After simulating a certain number of paths, we select the root node's n_0 child node n_1^* with the highest information value $I(\psi_{t+1})$ averaged over all path simulations through node n_1^*. The UAV moves to its associated position $\mathbf{p}_{t+1}^{n_1^*}$.

To show that our approach supports various planning algorithms, we proposed the four diverse planners above and demonstrate their integration into our modular framework. Furthermore, we highlight that our planning strategies are agnostic to the acquisition functions introduced in Sec. 6.1.1.

6.2 Experimental Evaluation

Our experiments evaluate our proposed method and investigate our claims made in this chapter. In Sec. 6.2.2, we show that our adaptive planning methods reduce

Table 6.1: Environment, sensor, and UAV mission parameters for the three datasets.

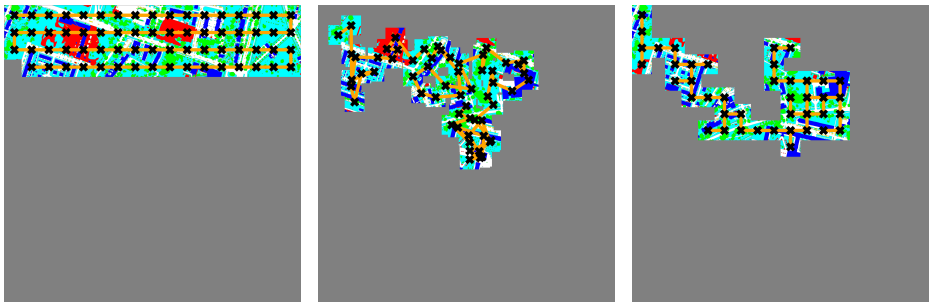
Parameter	Potsdam [68]	RIT-18 [75]	Flightmare [157]
Type	Orthomosaic	Orthomosaic	Unity/Gazebo
Classes	7	6	10
Task	Urban	Land Cover	Industrial
Area [m×m]	900×900	261×568	150×130
FoV [px×px]	400×400	400×400	480×720
GSD [cm/px]	15.0	8.0	8.3
Altitude [m]	30	15	20
Budget [s]	1800	400	150
Test Images	3500	3500	1000

the number of human-labelled images required to maximise semantic segmentation performance compared to pre-programmed data collection and state-of-the-art local planning proposed by Blum et al. [12]. We further verify this claim in Sec. 6.2.2.1 for various UAV-based semantic terrain mapping scenarios, and under task-dependent design choices, i.e. the UAV's starting position, pre-training scheme and model architecture, performing a sensitivity analysis in Sec. 6.2.4. Second, in Sec. 6.2.3.2, we validate that our probabilistic global mapping of gathered information enhances map-based planning performance. Third, we showcase that our Bayesian extension of the deterministic ERFNet [131] improves semantic segmentation performance and yields more consistent model uncertainty estimates, resulting in higher planning performance than previously used non-Bayesian objectives. To this end, we demonstrate the superior active learning performance of Bayesian over non-Bayesian objectives in Sec. 6.2.3.4, and confirm the superior model performance of our Bayesian ERFNet using Monte Carlo dropout and ensembles in Sec. 6.2.3.1 and Sec. 6.2.3.3 respectively.

6.2.1 Experimental Setup

Baselines. We compare our planning framework against three baselines: a traditionally-used coverage-based collection strategy [48], and two random walk-based exploration planners. The coverage strategy precomputes a static path maximising the area covered by the UAV to foster spatial diversity of training data. We precompute lawnmower-like patterns before a mission starts, alternate the pattern's orientations, and vary the step size between measurement positions.

We consider two random walk schemes, local and global. Similar to the local planner, the local random walk chooses for a given UAV position one of the four image edges at random. It follows the edge direction with a predefined step size. The global random walk randomly selects a UAV position in the continuous space above the terrain, similar to our map-based planners. For better budget



(a) Coverage pattern (b) Global random walk (c) Local random walk

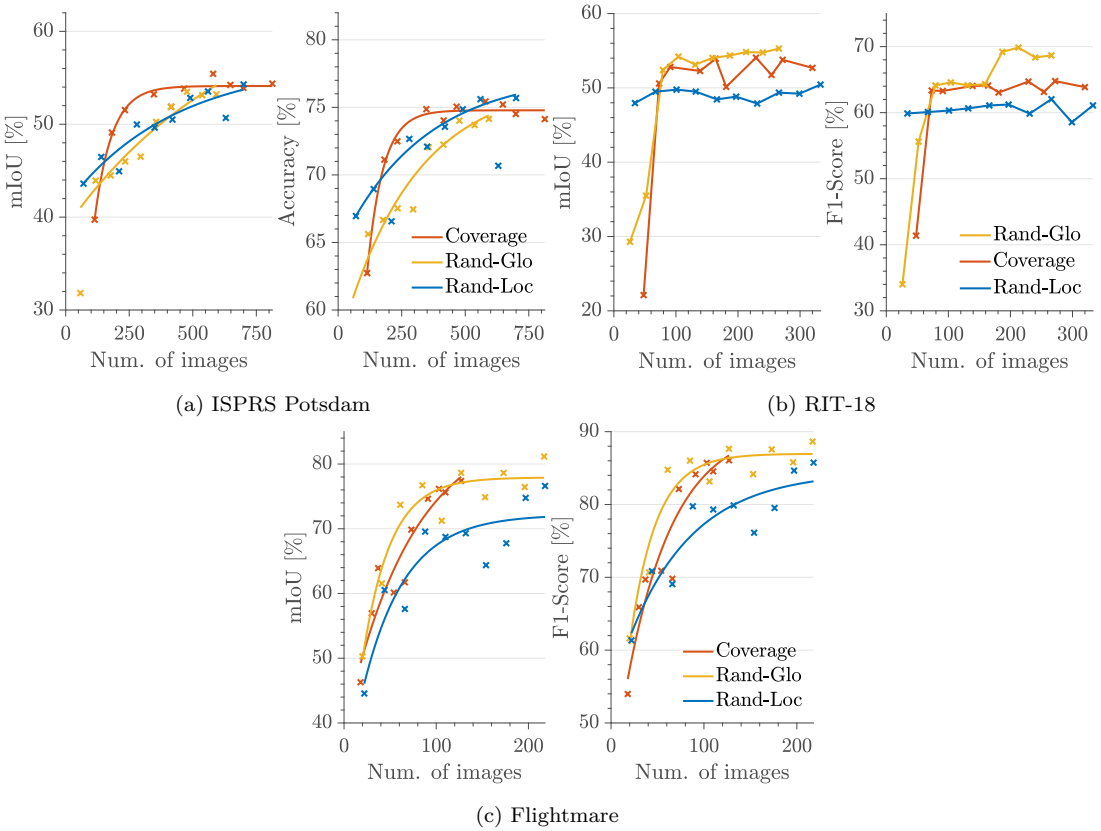
Figure 6.6: Examples of paths planned by the baseline strategies on ISPRS Potsdam [68]. Orange lines show paths planned in one mission, black crosses indicate collected training images, and gray areas depict unexplored terrain.

management, we sample positions uniformly at random between a minimum and maximum radius around the UAV. In this way, both random walks aim to foster data diversity while handling the budget properly. Both variants resemble the action spaces of the planners introduced in Sec. 6.1.3, allowing us to study the influence of the action space design and verify that our adaptive planners maximise active learning performance beyond random effects. Fig. 6.6 exemplifies the paths planned by all three baselines on the ISPRS Potsdam dataset [68].

Datasets. We evaluate our framework on two real-world orthomosaic datasets and in a photorealistic physics-based UAV simulator resembling real-world deployment conditions. Environment, sensor, and UAV mission settings are shown in Tab. 6.1. Below, we highlight the key differences between the three scenarios.

First, we use the large 7-class urban aerial ISPRS Potsdam orthomosaic dataset [68]. This dataset is characterised by a dense spatial distribution of classes, such that the coverage and random walk baselines can collect visually and semantically different features easily. We sample 4000 train, 1000 validation, and 3500 test images uniformly at random from non-overlapping areas in the orthomosaic. We use the ISPRS Potsdam dataset for the main planning experiments in Sec. 6.2.2 and for evaluating our mapping module in Sec. 6.2.3.2, Bayesian ensemble in Sec. 6.2.3.3, and planning objectives in Sec. 6.2.3.4.

Second, we use the land cover RIT-18 orthomosaic dataset [75] for our experiments in Sec. 6.2.2.1. It consists of semantics covering large connected areas, e.g. vegetation and lake, and covers smaller objects, e.g. buildings, with six classes in total. Since the RIT-18 dataset does not provide different orthomosaics for training and testing, we evaluate the UAV's vision capabilities by sampling the test set from the same area. In contrast to the ISPRS Potsdam dataset, this does not allow us to draw conclusions about the model's generalisability, but about its performance in the deployed environment, which is still a crucial skill for autonomous robots. Our RIT-18 evaluation resembles that of Blum et al. [12].



(a) ISPRS Potsdam

(b) RIT-18

(c) Flightmare

Figure 6.7: Comparison of active learning performance of our three baseline approaches with Monte Carlo dropout inference (a) on ISPRS Potsdam [68], (b) on RIT-18 [75], and (c) in the photo-realistic Flightmare UAV simulator [157]. Steeper curves indicate better active learning performance. We compare our planning strategies to the best-performing baseline in each setting: coverage on ISPRS Potsdam and the global random walk on RIT-18 and in Flightmare.

Last, we evaluate our framework in Sec. 6.2.2.1 using Flightmare, a photorealistic simulator emulating UAV dynamics [157]. We deploy a UAV in the provided 'Industrial' environment introducing 10 semantic classes of different spatial distributions, e.g. hangar, container, road, fence, and pipe. The scene covers a dense area leading to compactly distributed semantics easily explorable by the baseline approaches. As the 'Industrial' terrain is small, we evaluate the UAV's semantic segmentation performance in the deployed environment only.

We perform a study comparing the active learning performance of the baselines in Fig. 6.7. On the ISPRS Potsdam dataset, the coverage pattern is the superior baseline. The global random walk exploration performs best on the RIT-18 dataset and in the Flightmare simulator. While Monte Carlo dropout is used in Fig. 6.7 to predict semantic segmentation, we found that similar results hold true for deterministic network and ensemble inference. For visual clarity, we only compare our framework to the baselines with the strongest performance.

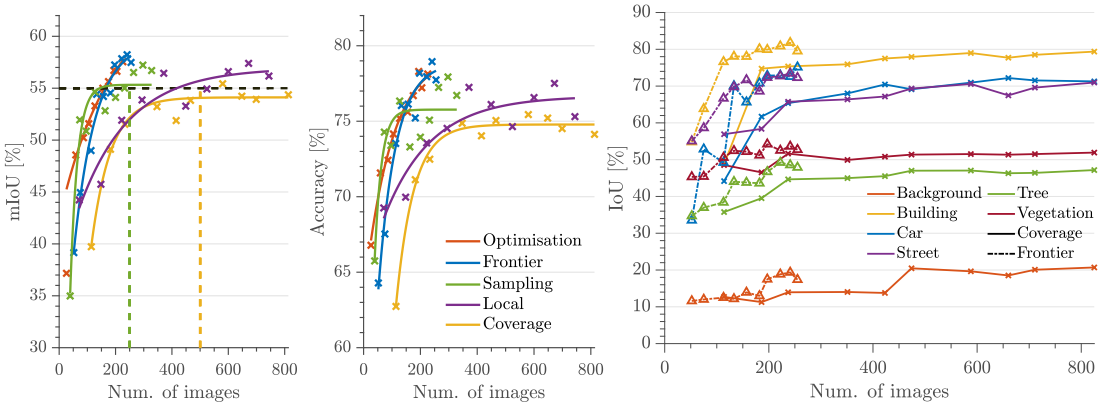
Evaluation Metrics. Our planning pipeline aims to maximise semantic segmentation performance with minimal human labelling effort, i.e. minimal

number of human-labelled images. In line with the standard in active learning literature [8, 12, 47, 53, 65, 70, 147, 165, 166, 173, 183], our key evaluation metrics assess semantic segmentation performance (dependent variable) over the number of collected training images (independent variable). Higher semantic segmentation performance thanks to newly added images indicates better active learning and thus indicates better planning performance. We choose mIoU, per-pixel accuracy, and per-pixel F1-score to access semantic segmentation performance. The mIoU metric is used in semantic segmentation benchmarks [31, 42] and defined as $\text{mIoU} = \frac{TP}{TP+FP+FN}$, where TP, FP, TN, and FN are the true and false positives, and true and false negatives. Per-pixel accuracy acc and F1-score f1 are typically used in classification benchmarks [33]. They are defined as $\text{acc} = \frac{TP}{TP+FP+TN+FN}$ and $\text{f1} = \frac{2TP}{2TP+FP+FN}$. RIT-18 and Flightmare have strongly imbalanced class distributions. Thus, we use the F1-score instead of accuracy for these scenarios. Training datasets are incrementally collected while exploring an unknown environment. Hence, the training image distribution changes during deployment as new visual features or semantics are discovered. This leads to non-monotonic model improvements as the training distribution could differ from the true distribution. To make model performance trends easier to follow, we additionally fit trend lines for the experiments conducted on ISPRS Potsdam and Flightmare. As performance trends are less regular on RIT-18 due to the more challenging exploration of semantics, we show piecewise linear line plots for these experiments.

Training Procedure. We use our lightweight Bayesian ERFNet for semantic segmentation proposed in Sec. 6.1.1. The model is pre-trained on the Cityscapes dataset [31]. We start experiments and training after each of the 10 subsequent data collection missions from the same checkpoint. This also avoids catastrophic forgetting and accumulating train time. We re-train the model until convergence with batch size 8 and weight decay $\lambda = (1-p)/2N$ in Eq. (6.1), where $p = 0.5$ is the dropout probability, and N is the number of training images as in [47]. All other hyperparameters follow ERFNet [131], not tuned for maximal performance in our setting, and kept fixed with changing datasets and planners.

Planning Hyperparameters. Our optimisation-based planner leverages the CMA-ES procedure as it has been shown to yield competitive performance in terrain monitoring tasks [61, 126]. We fix a set of hyperparameters for all planners with reasonable length scales on ISPRS Potsdam, i.e. UAV step sizes, minimum and maximum action space radii, grid discretisation, and initial CMA-ES covariance. Only these hyperparameters, which depend on the aerial dimensions, are scaled with changing environment sizes. The scale-independent hyperparameters, e.g. the number of tree search simulations, are set in line with prior works [13, 126]. We fix the UAV's starting position to the top-left terrain corners.

Planning Strategies. We outline our planning strategies in detail in Sec. 6.1.3.



(a) Overall active learning performance of different planners (b) Frontier planner per-class performance

Figure 6.8: (a) Comparison of active learning performance with a Bayesian model uncertainty-based planning objective estimated by Monte Carlo dropout and computing informative prior maps before each mission starts. All adaptive planners outperform the coverage baseline (yellow) with less training data as shown by the dashed lines. Our map-based planners outperform the local planner (purple). (b) Comparison of per-class performance of map-based frontier vs. coverage planning. The frontier planner outperforms the coverage baseline (yellow) in almost all classes as our framework captures task-dependent inter- and intra-class model uncertainties.

In our experiments, we refer to the planners in the legends as follows: the local planner is named *Local*, the frontier-based planner is named *Frontier*, the optimisation-based planner is named *Optimisation*, and the sampling-based planner is named *Sampling*. The baseline approaches are referred to as follows: the coverage pattern is called *Coverage*, and the global and local random walk exploration strategies are abbreviated with *Rand-Glo* and *Rand-Loc* respectively.

6.2.2 Adaptive Path Planning for Active Learning

The first set of experiments analyses the performance of our adaptive planning approach. It (i) verifies the superior active learning performance of our planning framework over the baselines, and (ii) shows that our global map-based planners outperform state-of-the-art local planning, supporting our first claim. The experiments are evaluated on the ISPRS Potsdam dataset [68] with recomputed map priors before each mission starts and using the Bayesian model uncertainty planning objective estimated by Monte Carlo dropout.

Fig. 6.8a summarises the active learning performance with the informed mapping strategy for each planner. All adaptive planners reach higher final prediction performance than the coverage baseline (yellow). This supports the claim that our framework is generally applicable to different planning algorithms, and it also suggests that adaptive replanning is key to efficiently improving robot vision. Notably, our global map-based planners (orange, blue, green) exceed the coverage baseline's maximum prediction performance ($\approx$ 55% mIoU, black dashed line)

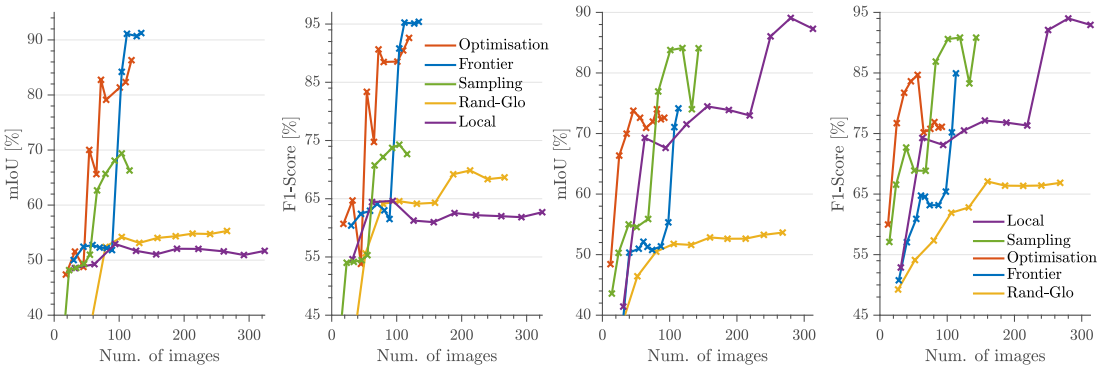
after ≤ 250 labelled images (dashed green line), while the baseline requires ≈ 500 labelled images (yellow dashed line) to reach this performance. Our map-based planners show stronger active learning performance than the local planner (purple) proposed by Blum et al. [12], particularly for the uncertainty-based objective. The map-based planners' performances upper-bound the local planner's performance for any number of labelled images. Map-based planners drastically reduce training data requirements and tend to achieve higher final performance.

To better understand the benefits of our adaptive planning method, Fig. 6.8b exemplarily compares the per-class active learning performance of the map-based frontier planner (dashed lines) to the coverage baseline (solid lines) in the ISPRS Potsdam scenario. Our adaptive frontier-based planning strategy shows higher active learning performance in almost all classes, irrespective of their training data support. Interestingly, the 'car' class (blue) has lower training data support than the 'tree' (green) and 'vegetation' (red) classes but shows stronger IoU performance, even with non-targeted coverage planning. However, adaptive planning improves the 'car' prediction performance even faster than the non-targeted baseline, showing the benefit of our framework for classes with little training data support. Furthermore, the 'tree', 'background' (orange), and 'vegetation' classes have high training data support but are difficult to distinguish. Their visual appearance from a top-down view depends on the image resolution, altitude, and season. This leads to challenging predictions, which may be partially attributed to data instead of model uncertainty, and which cannot be explained away with more training data [76]. Thus, not all classes with high training data support benefit to the same extent from adaptive planning. At the same time, the 'building' class (yellow) has high training data support and is reliably detected by both planners. The frontier-based planner still shows faster performance improvement as our framework can account for the differing visual appearance and geometry of office buildings, historical buildings, and townhouses. Overall, the results suggest that our adaptive map-based planning accounts for task-dependent inter-class and intra-class model uncertainties, leading to superior active learning performance.

6.2.2.1 Other Semantic Terrain Mapping Scenarios

The second set of experiments suggests that (i) our planning framework reduces the number of labelled images required to maximise segmentation performance across substantially different environments, and (ii) our global map-based planning strategies outperform state-of-the-art local planning in most cases, irrespective of the chosen planning objective, further validating our first claim.

We evaluate our framework on the RIT-18 dataset [75] and in the Flightmare simulator [157]. The RIT-18 semantics cover large areas, leading to challenging exploration. The Flightmare simulator resembles real-world UAV control over



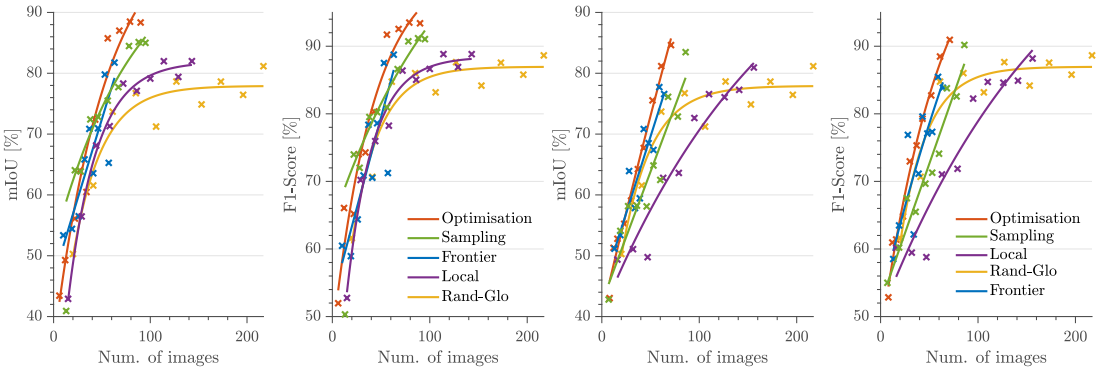
(a) Bayesian model uncertainty-based objective (b) Representation-based novelty objective

Figure 6.9: Active learning results on the RIT-18 dataset [75] using informative prior maps with (a) the Bayesian model uncertainty objective estimated by Monte Carlo dropout, and (b) the representation novelty objective. All map-based planners significantly outperform the random walk baseline (yellow). Our map-based planners lead to substantially higher active learning performance than the local planning strategy (purple) using the model uncertainty objective.

a compact, easy-to-explore photorealistic industrial terrain with strong random walk baseline performance. We access the framework's performance using the Bayesian model uncertainty estimated with Monte Carlo dropout, see Eq. (6.3), and the representation novelty score given by Eq. (6.4).

Fig. 6.9 summarises our planning results on the RIT-18 dataset [75]. Semantics cover large areas, leading to challenging exploration influencing the training data class distribution. Thus, we see non-monotonic model performance improvements on RIT-18. All map-based planners show significantly higher final segmentation performance than the random walk (yellow), irrespective of the planning objective. This confirms that adaptive planning reduces human labelling effort over vastly differing terrains. Particularly, our map-based planners require fewer training images to achieve segmentation performance on par or higher than the local planner (purple) in most cases. Notably, the local planner performs worse than the baseline using Bayesian model uncertainty. This shows that our map-based planners are more generally applicable than the local planner.

Fig. 6.10 illustrates our results in the Flightmare simulator [157]. All planners using the Bayesian model uncertainty objective display higher active learning performance than the random walk baseline (yellow). Using the representation-based objective, only our two map-based optimisation (orange) and sampling (green) planners result in higher final prediction performance than the baseline. Combined with the RIT-18 results in Fig. 6.9, this suggests that our Bayesian model uncertainty-based objectives are more robustly applicable across varying terrains compared to the representation novelty score proposed by Blum et al. [12]. One possible explanation could be that Bayesian model uncertainty is more strongly correlated with prediction errors, as indicated by our results in Sec. 6.2.3.1 and



(a) Bayesian model uncertainty-based objective

(b) Representation-based novelty objective

Figure 6.10: Active learning results in the Flightmare simulator [157] using informative prior maps with (a) the Bayesian model uncertainty objective (Eq. (6.3)), and (b) the representation novelty objective (Eq. (6.4)). All planners outperform the random walk baseline (yellow) using the Bayesian model uncertainty objective. Using the representation novelty objective, only our map-based optimisation and sampling planners show higher final prediction performance.

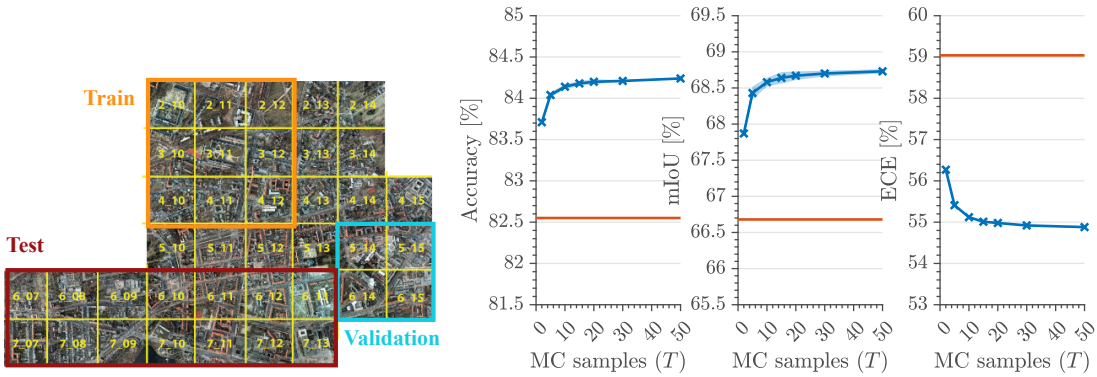
Sec. 6.2.3.4. In most cases, our map-based planners show higher active learning performance in both terrains than local planning. This verifies that our map-based planners are crucial for informative data collection, while local planning is not robustly applicable to varying terrains and planning objectives.

6.2.3 Ablation Studies

The third set of experiments is a series of ablation studies on our planning framework to assess the second and third claim made in this chapter. In Sec. 6.2.3.1, we verify that our Bayesian ERFNet with Monte Carlo dropout achieves higher semantic segmentation performance than the standard ERFNet and provides consistent model uncertainties. Similarly, in Sec. 6.2.3.3, we show that our Bayesian ensemble provides reliable uncertainty estimates for active learning planning objectives and achieves higher prediction performance than non-Bayesian and Bayesian ERFNet with Monte Carlo dropout. Additionally, in Sec. 6.2.3.4, we validate that our framework supports various active learning acquisition functions and yields superior active learning performance using Bayesian instead of previous non-Bayesian planning objectives. These results validate our third claim. Moreover, we demonstrate the benefit of our mapping module for the map-based planners in Sec. 6.2.3.2, supporting our second claim.

6.2.3.1 Bayesian ERFNet via Monte Carlo Dropout

The first ablation study results aim to show that our Bayesian ERFNet proposed in Sec. 6.1.1 achieves higher semantic segmentation performance than the standard ERFNet [131] and provides consistent model uncertainty estimates for



(a) ISPRS Potsdam dataset

(b) Bayesian ERFNet with Monte Carlo dropout performance

Figure 6.11: (a) The urban ISPRS Potsdam orthomosaic [68]. We simulate images and labels with a square footprint, downwards-facing camera, and ground sample distance of 15 cm/px to generate training (orange), validation (blue), and test (red) data from disjoint areas. (b) Our Bayesian ERFNet (all, $p = 50\%$, blue) with varying Monte Carlo dropout samples compared against non-Bayesian ERFNet (all, $p = 50\%$) (orange) on ISPRS Potsdam. Metrics are averaged over three trials with shaded regions indicating standard deviations. For $T = 50$, Bayesian ERFNet improves mIoU by 4.6% (left, middle) and reduces ECE by 7.6% (right).

active learning. We perform an ablation study with varying Bayesian units of the ERFNet base architecture to find the best-performing trade-off between prediction performance and consistent model uncertainty estimation. We assess four different Bayesian variants of ERFNet with varying dropout probabilities $p = \{10\%, 30\%, 50\%\}$:

- **Standard**: Dropout layers after all non-bottleneck-1D layers in the encoder as in the normal ERFNet implementation [131].
- **Center**: Dropout layers after the last four and first two encoder and decoder non-bottleneck-1D layers respectively.
- **Classifier**: Single dropout layer after the last decoder non-bottleneck-1D layer before the classification head.
- **All**: Dropout layers after all non-bottleneck-1D layers in the encoder and decoder as proposed in Sec. 6.1.1.
- **Non-Bayesian**: Standard ERFNet architecture without Monte Carlo dropout but a single deterministic forward pass at inference as proposed in [131].

All models are trained and evaluated on the 7-class urban aerial ISPRS Potsdam orthomosaic dataset [68]. We simulate images and labels at random uniformly chosen positions from 30 m altitude with a square footprint, downwards-facing camera, and ground sample distance of 15 cm/px. In total, we create 4000, 1000, and 3500 training, validation, and test images, respectively, from disjoint

Table 6.2: Ablation study of our Bayesian ERFNet trained with varying dropout layers and probabilities $p = \{10\%, 30\%, 50\%\}$ on the ISPRS Potsdam dataset [68] with $T = 50$ MC dropout samples at inference. Our best Bayesian ERFNet (all, $p = 50\%$) outperforms the best non-Bayesian ERFNet ($p = 50\%$) by 2.5% mIoU and 4.6% ECE.

Layer variant	Accuracy [%] ↑	mIoU [%] ↑	ECE [%] ↓
Non-Bayesian [131]	82.28 / 81.99 / 82.71	64.47 / 64.35 / 66.24	59.98 / 59.09 / 59.43
Standard	82.47 / 82.21 / 83.94	64.81 / 64.70 / 68.00	57.32 / 55.46 / 55.41
Center	81.58 / 82.26 / 82.91	62.34 / 64.47 / 65.78	60.05 / 57.61 / 58.55
Classifier	80.80 / 82.14 / 81.04	62.02 / 63.94 / 62.16	60.62 / 60.40 / 61.02
All (Sec. 6.1.1)	84.00 / 82.20 / **84.24**	67.75 / 63.93 / **68.74**	55.76 / **53.26** / 54.87

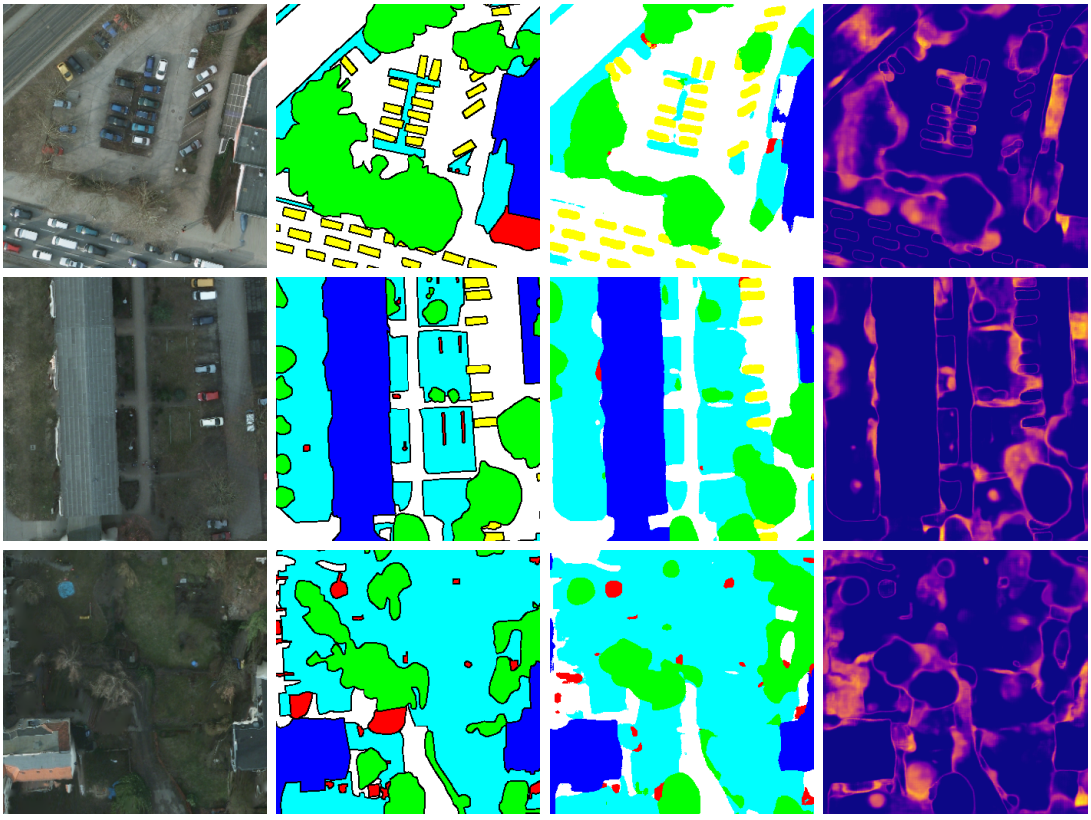
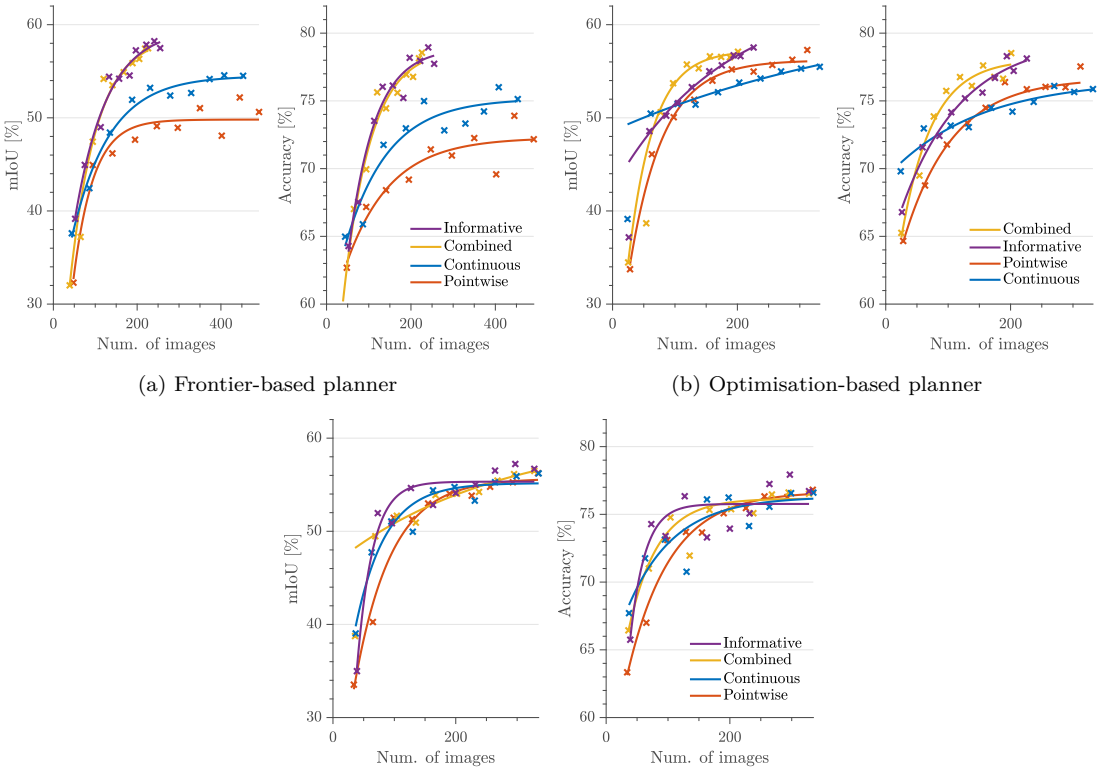


Figure 6.12: Qualitative results of Bayesian ERFNet with Monte Carlo dropout (all, $p = 50\%$) trained on the ISPRS Potsdam dataset [68]. High model uncertainty estimates (yellow) in misclassified regions are potentially valuable to guide the UAV during an active learning mission.

areas as depicted in Fig. 6.11a. We assess the model uncertainty estimation quality by the expected calibration error (ECE) [52]. At test time, we use a reasonably large number of Monte Carlo dropout samples of $T = 50$.

Tab. 6.2 summarises our results. With highest accuracy and mIoU, Bayesian ERFNet-All, proposed in Sec. 6.1.1, trained with $p = 50\%$ performs best. No-



(a) Frontier-based planner

(b) Optimisation-based planner

(c) Sampling-based planner

Figure 6.13: The planners consistently benefit from recomputing informative map priors before a mission starts (purple). The performance gain of mapping a continuous sensor stream (blue) instead of only mapping images at planned measurement positions (orange) is less significant. Combining both (yellow) also leads to consistent performance improvements. Our informative mapping approach drastically improves the greedy frontier-based strategy in particular.

ticeably, our Bayesian ERFNet-All ($p = 50\%$) outperforms its strongest non-Bayesian counterpart ($p = 50\%$) by 2.5% mIoU while resulting in 4.6% improved ECE. Qualitatively, Fig. 6.12 confirms high uncertainty in misclassified, cluttered regions. Hence, our Bayesian ERFNet provides a reliable planning objective for active learning. These results validate that our probabilistic model interpretation improves performance and provides reliable uncertainty estimates.

To assess computational requirements and the applicability of our method for online replanning, we study the performance of our Bayesian ERFNet with varying numbers of Monte Carlo dropout samples $T = \{2, 5, 10, 15, 20, 30, 50\}$ in Fig. 6.11b. As the number of MC dropout samples increases, prediction performance and ECE both improve. Favourably for adaptive online robotic decision-making, $T \approx 20$ samples are already sufficient for converging performance gains.

6.2.3.2 Informative Mapping

To support the claim that our new mapping module described in Sec. 6.1.2 is important for planning performance, we perform an ablation study to measure

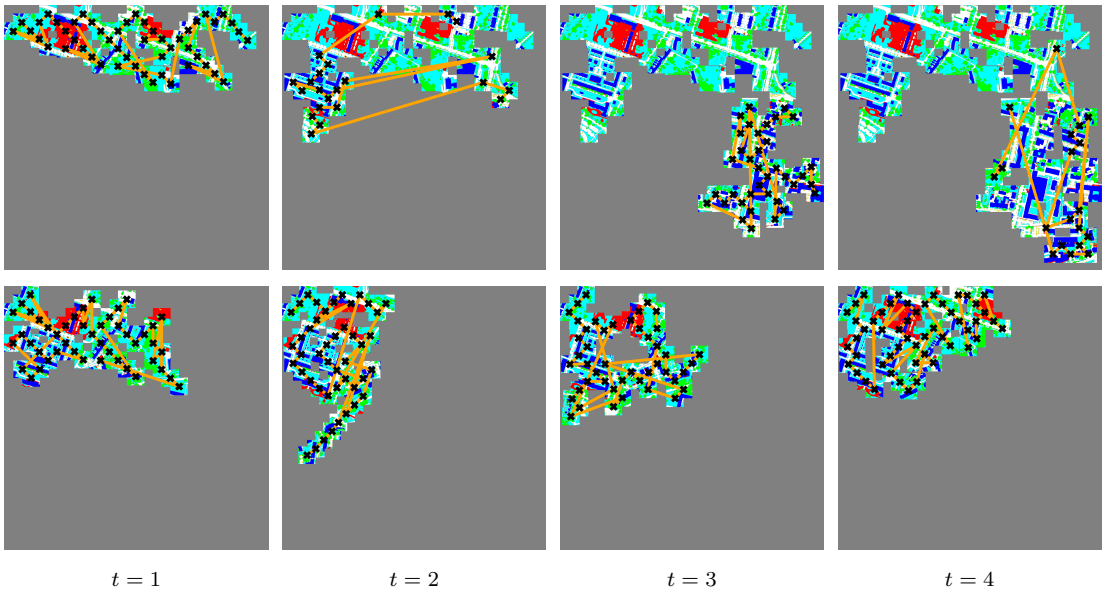


Figure 6.14: Examples of paths planned on ISPRS Potsdam using the frontier-based planning strategy with (top) and without (bottom) our approach for precomputing informative prior maps. The priors are computed before each of four subsequent missions, with the UAV starting in the top-left corner. As shown by the planned paths (orange lines) and measurement positions (black crosses), using informative priors facilitates spatial exploration across missions and leads to more targeted training data collection within missions.

its effect on our map-based planners. We consider two mapping setups where the UAV either maps training images at planned measurement positions only (pointwise sensor) or maps the images continuously as it moves (continuous sensor stream). Fig. 6.13 displays the active learning performance of our map-based planners (i) recomputing informative prior maps before each mission starts based on previously collected data and the re-trained network (purple), (ii) mapping a continuous RGB image stream (blue) instead of mapping training images at planned measurement positions only (orange), and (iii) combining both informative prior maps and mapping continuous sensor streams (yellow). The experiments are evaluated on the ISPRS Potsdam dataset [68]. We fix Bayesian model uncertainty as our planning objective estimated with Monte Carlo dropout.

All map-based planners show better performance with recomputed map priors as they exploit already collected terrain information. This suggests that mapping and updating knowledge across missions with re-trained networks, i.e. changing vision capabilities, is key to strong planning performance. In contrast, mapping more information during a mission with a fixed network is less crucial. Mapping a continuous image stream instead of mapping at planned measurement positions only leads to performance improvements for the greedy frontier-based planner, while non-greedy planners do not benefit from mapping more information. Accordingly, combining mapping continuous sensor streams and recomputing map

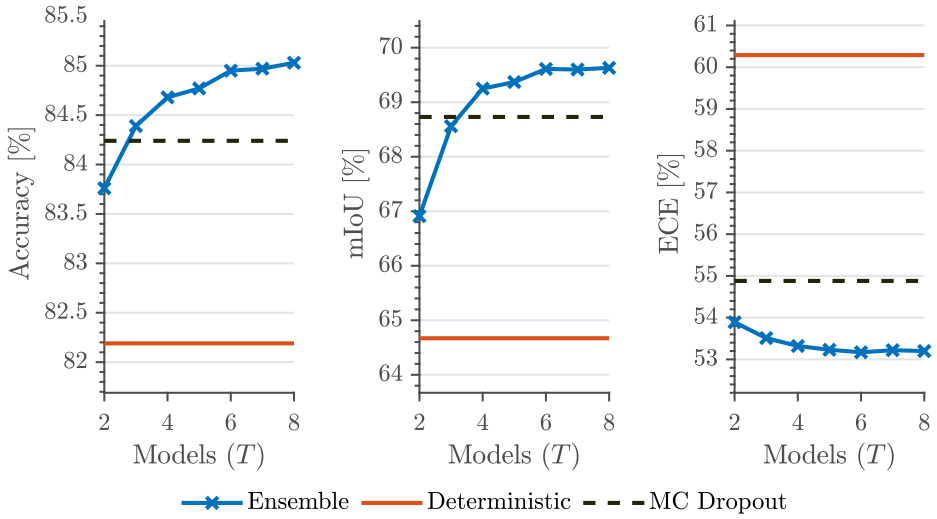


Figure 6.15: Performance of our Bayesian ensemble with varying number T of ERFNets (blue), non-Bayesian deterministic ERFNet [131] (orange), and Bayesian ERFNet with $T = 50$ Monte Carlo (MC) dropout samples (black, dashed) on ISPRS Potsdam. For $T = 8$, the ensemble improves mIoU by 4.96% (middle) and reduces ECE by 7.09% (right) over the deterministic ERFNet, and improves mIoU by 0.90% and reduces ECE by 1.68% over Monte Carlo dropout.

priors leads to higher performance of the frontier- and optimisation-based planner. The sampling-based planner does not show a performance gain when combining mapping continuous sensor streams and recomputing map priors. Particularly, our frontier-based planner shows significant improvements by leveraging the informative mapping procedure. Qualitatively, Fig. 6.14 verifies that informative prior maps for frontier-based planning lead to more efficient terrain exploration across missions and targeted data collection within missions, resulting in higher model performance with fewer training images. Our optimisation- and sampling-based planners are more robust to less informed map priors, the reason being perhaps because they use non-myopic planning along multiple waypoints, while the frontier-based planner only reasons about the next waypoint.

6.2.3.3 Bayesian ERFNet Ensemble Study

To confirm that our Bayesian ERFNet ensemble delivers informative model uncertainties and yields superior performance, we train it on the ISPRS Potsdam dataset [68] and compare it to our Bayesian ERFNet with Monte Carlo dropout and the deterministic ERFNet proposed by Romera et al. [131].

To assess our Bayesian ensemble's prediction capabilities and computational efficiency for online inference on UAVs, we study its performance with varying numbers of ERFNet models $T = \{2, \ldots, 8\}$ in Fig. 6.15. We compare the ensemble's performance (blue) to the deterministic ERFNet [131] (orange) using a single forward pass and to our Bayesian ERFNet using $T = 20$ Monte Carlo

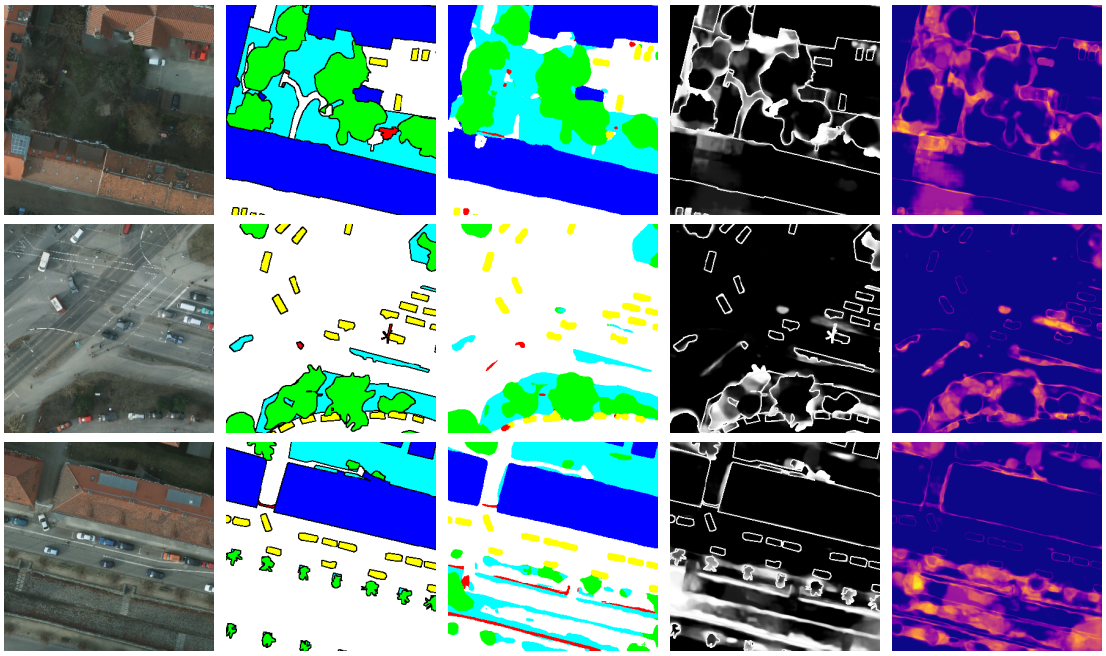


Figure 6.16: Qualitative results with our ensemble of $T = 8$ ERFNets trained on the ISPRS Potsdam dataset [68]. High model uncertainty (yellower) in misclassified regions (whiter, higher negative log-likelihood) validates that our Bayesian ensemble provides consistent model uncertainty estimates as a basis for an adaptive planning objective in our framework.

dropout samples (black, dashed) for converging to maximal performance. To quantify the reliability of estimated uncertainties, we measure model calibration using the ECE metric [52]. Intuitively, model calibration is high, i.e. ECE is low, when the model's probabilistic predictions match its accuracy on a test set.

For $T = 8$ models, our ensemble (blue) improves performance by 4.96% mIoU and ECE by 7.09% over the deterministic ERFNet. Additionally, for $T = 8$ models, our ensemble improves performance by 0.90% mIoU and ECE by 1.68% compared to the Bayesian ERFNet with Monte Carlo dropout. Overall, as the number of models increases, both performance and calibration improve. Favourably for online inference, performance gains already converge with $T \approx 6$ models. Moreover, the Bayesian ensemble performs on par with the Bayesian ERFNet ($T = 20$ Monte Carlo dropout samples) already with $T = 3$ ERFNet models. Thus, our ensemble requires approx. $6\times$ fewer forward passes, i.e. compute resources, at deployment to achieve the same performance. At train time, the ensemble's compute requirements scale linearly with the number of models T, while the Monte Carlo dropout Bayesian ERFNet's are constant. However, training is performed offline and is thus not time-critical. For details about efficient ensemble training, we refer to Huan et al. [67]. Qualitatively, Fig. 6.16 verifies high model uncertainty in misclassified or hard-to-predict regions. Thus, the ensemble's model uncertainties provide reliable information for planning objectives.

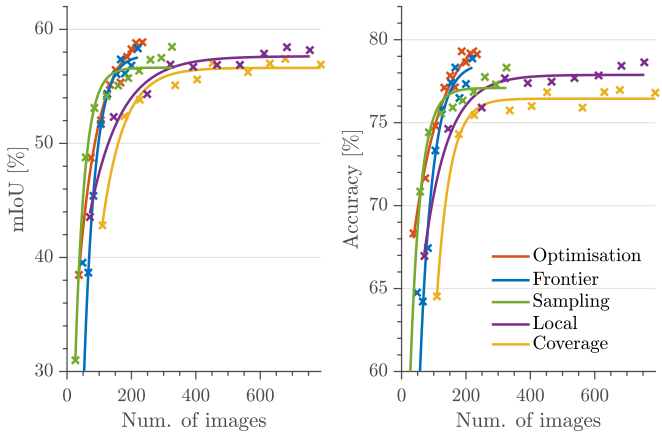


Figure 6.17: Comparison of active learning performance with a Bayesian model uncertainty-based planning objective estimated by an ensemble of $T = 4$ ERFNets and computing informative prior maps before each mission starts. All adaptive planners outperform the coverage baseline. Our global map-based planners outperform the local planner with less training data.

6.2.3.4 Comparison of Planning Objectives

Our third set of ablation study results shows that Bayesian model uncertainty-based planning objectives guarantee a strong active learning performance irrespective of the uncertainty estimation technique. Furthermore, it verifies that our adaptive planning supports various active learning acquisition function paradigms, including representation-based and uncertainty-based objectives.

As we show in this experiment, Bayesian model uncertainty-based planning objectives outperform baselines with different uncertainty estimation techniques. We investigate our Bayesian ensemble's active learning performance on ISPRS Potsdam. For a fair assessment, we evaluate the coverage baseline with ensemble inference. Fig. 6.17 summarises the results using our Bayesian ensemble of $T = 4$ ERFNets for all planning approaches. All adaptive planners show better active learning performance than the coverage baseline (yellow). This confirms that adaptive planning benefits from Bayesian model uncertainty-based objective functions. Similar to our Monte Carlo dropout-based uncertainty estimation in Fig. 6.8a, map-based planners (orange, blue, green) achieve higher prediction performance with fewer training images compared to the local planner (purple). This illustrates the advantage of our map-based planners over local planning.

To further support our framework's generality under various uncertainty-based objective functions, we investigate its performance using a classical non-Bayesian entropy-based acquisition function [47, 128]. Given an image $\mathbf{z}$ and a model with deterministic parameters $\mathbf{w}$, the prediction $p(\mathbf{y} \mid \mathbf{z}, \mathbf{w})$ is the likelihood estimate over semantic labels $\mathbf{y}$. Then, the prediction entropy

$$\mathbb{H}(\mathbf{y}) = -p(\mathbf{y} \mid \mathbf{z}, \mathbf{w})^{\top} \log\big(p(\mathbf{y} \mid \mathbf{z}, \mathbf{w})\big) \tag{6.13}$$

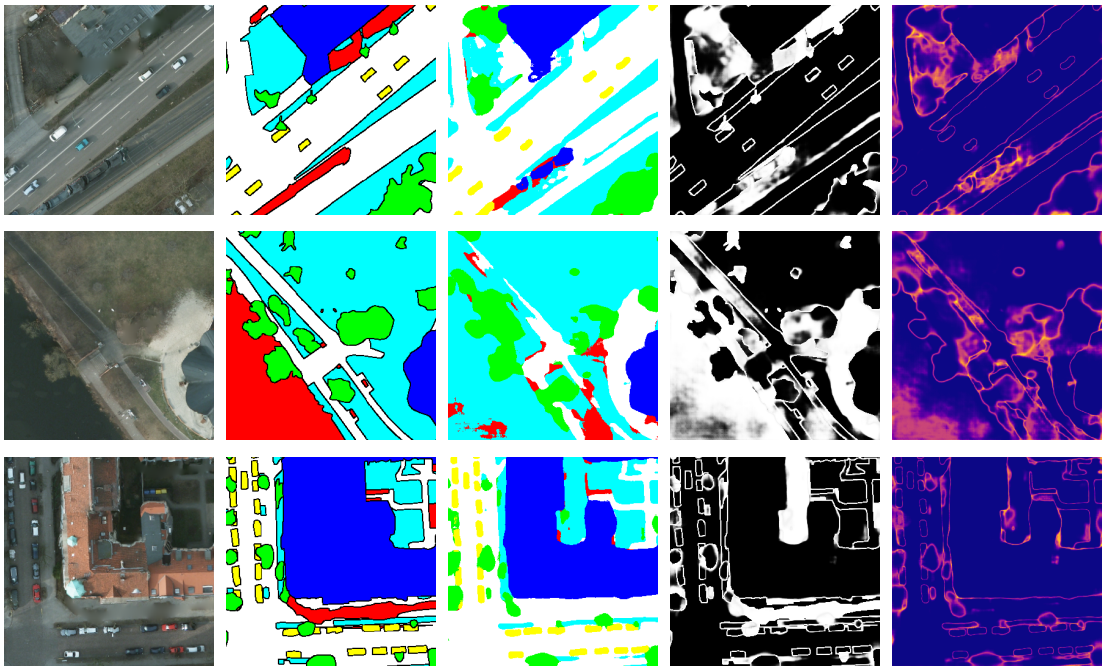
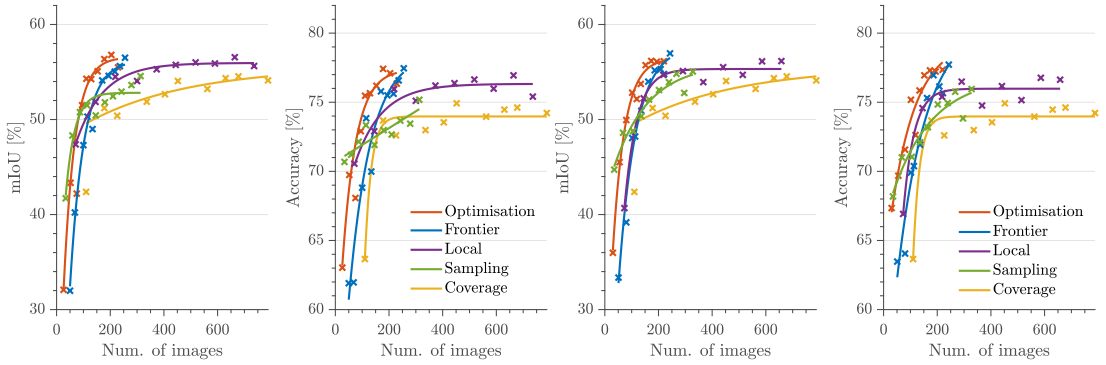


Figure 6.18: Qualitative results of the non-Bayesian deterministic ERFNet [131] trained on ISPRS Potsdam. Columns from left to right: RGB input, ground truth, prediction, error image (negative log-likelihood loss), prediction entropy. High uncertainty areas (yellower) correlate only weakly with areas of high prediction errors (whiter).

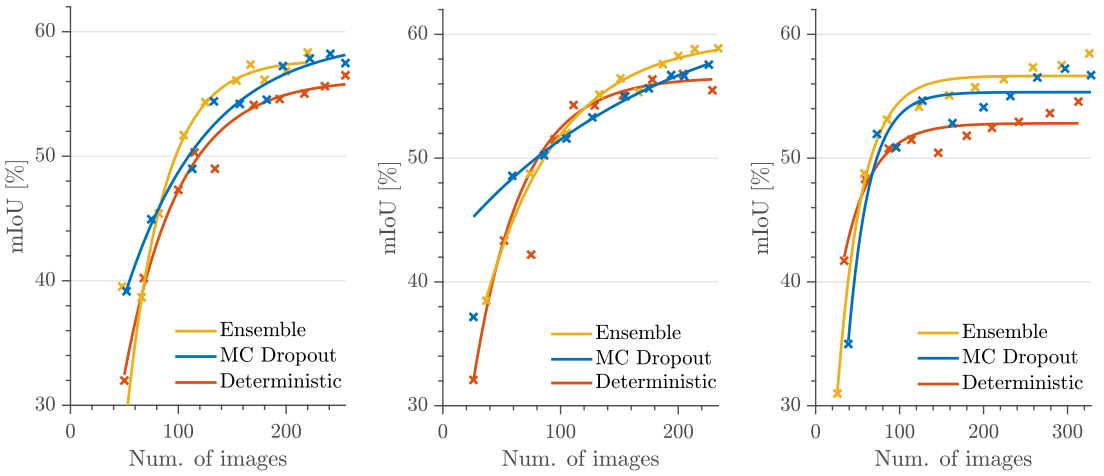


(a) Entropy-based planning objective

(b) Representation-based planning objective

Figure 6.19: (a) Comparison of active learning performance with a non-Bayesian entropy-based planning objective and computing informative prior maps before each mission starts. The frontier-based, optimisation-based, and local planner outperform the coverage baseline's active learning performance. (b) Comparison of active learning performance with representation-based novelty objective and computing informative prior maps before missions start. All adaptive planners outperform the baseline (yellow) with fewer training images.

is highest when the prediction is uniform, i.e. most uncertain. Qualitatively, Fig. 6.18 shows that non-Bayesian entropy only weakly correlates with prediction errors as it fails to estimate globally calibrated uncertainties. We replace the Bayesian model uncertainty, see Eq. (6.3), with the entropy of a deterministic



(a) Frontier (b) Optimisation (c) Sampling

Figure 6.20: Comparison of active learning performance of uncertainty-based planning objectives on ISPRS Potsdam. Our Bayesian uncertainty-based objectives (blue, yellow) tend to perform better than the non-Bayesian entropy-based objective (orange). The Bayesian ensemble (yellow) achieves the highest active learning performance across the planning strategies.

forward pass. For a fair comparison, the coverage baseline uses a deterministic forward pass as well. As shown in Fig. 6.19a, the optimisation-based, frontier-based and local planners outperform the baseline, while the sampling-based planner performs similarly to the baseline. In line with results for Bayesian model uncertainty-based objectives, the optimisation-based and frontier-based global planners show high prediction performance with substantially fewer training images compared to the local planning strategy. These results verify that our framework also supports non-Bayesian uncertainty-based objectives.

Fig. 6.20 compares the effect of non-Bayesian entropy-based (orange) and Bayesian model uncertainty-based planning objectives estimated by either Monte Carlo dropout (blue) or an ensemble (yellow) on the planners' performances. Particularly, the map-based planners achieve higher active learning performance using Bayesian model uncertainty-based objectives irrespective of the uncertainty estimation technique. The non-Bayesian uncertainty objective yields competitive performance with multiple planners in early missions. However, the Bayesian ensemble method generally leads to the best results. This could be due to two reasons. First, the ensemble shows higher prediction power than the non-Bayesian model as illustrated in Fig. 6.15. Second, our results in Fig. 6.18 and Fig. 6.15 suggest that non-Bayesian uncertainty is weakly calibrated. This could result in a less informative planning objective for training data collection.

To confirm that our framework applies to representation-based acquisition functions, we use the novelty score shown in Eq. (6.4) of a deterministic ERFNet in our planning objective. For a fair assessment, we also use a deterministic ERFNet for the coverage baseline. Qualitatively, Fig. 6.21 visualises the nov-

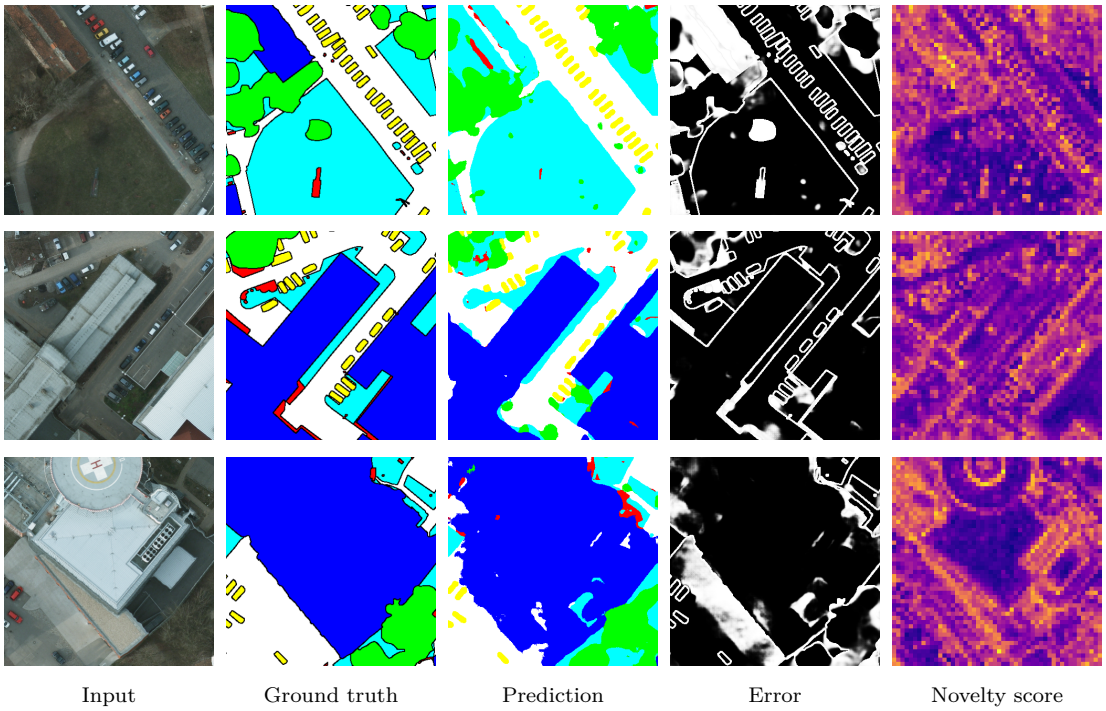


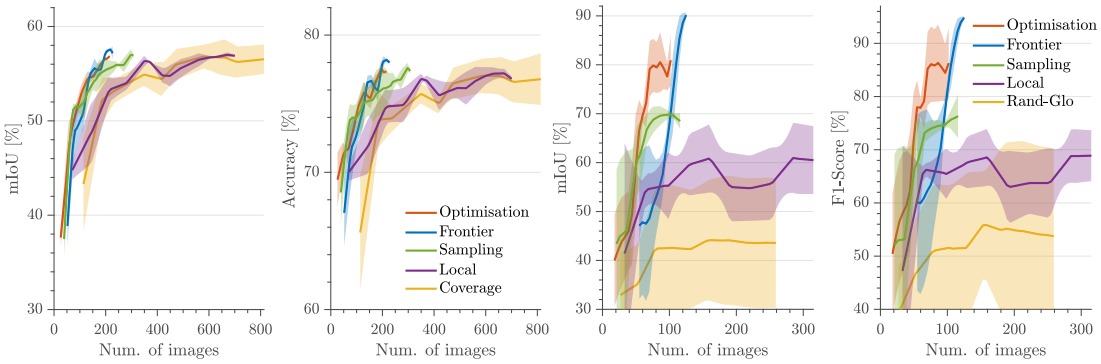
Figure 6.21: Qualitative results of a deterministic ERFNet [131] trained on ISPRS Potsdam [68]. High novelty scores (yellower) in case of rare visual cues, such as the helipad (bottom row), suggest that our representation-based objective provides useful information for planning.

elty scores of a network trained and tested on disjoint areas of ISPRS Potsdam. Although the novelties do not correlate strongly with prediction errors (whiter), high novelty (yellower) is assigned to rare visual cues, such as the helipad (bottom row), which could be an informative objective to collect diverse training images.

Fig. 6.19b depicts the active learning results using representation novelties in the planning objective. All adaptive planners achieve higher performance than the coverage baseline (yellow). Moreover, our map-based optimisation (orange) and frontier (blue) planners require fewer training images than the local planner (purple) to reach high performance. This validates our claim that our framework generally supports various acquisition function paradigms. Additionally, it ensures higher performance than the baseline approaches, irrespective of the planning objective. Our experiments suggest that the map-based planners outperform the local planner more significantly using Bayesian uncertainty objectives. This could be a result of the better-calibrated Bayesian uncertainty estimates as shown in Fig. 6.15, leading to more informative map-based planning objectives.

6.2.4 Sensitivity Analysis

The last set of experiments analyses our planning framework under various task-dependent design choices to further support our first claim made in this chapter.



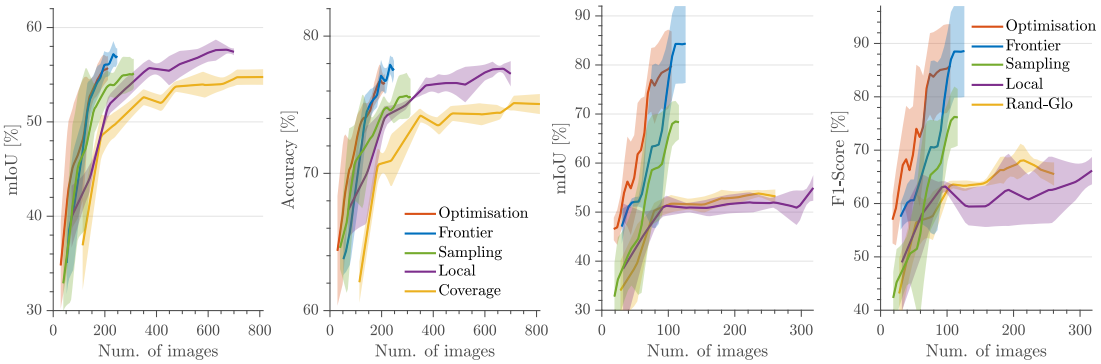
(a) ISPRS Potsdam dataset (b) RIT-18 dataset

Figure 6.22: Comparison of active learning performance on the (a) ISPRS Potsdam dataset [68] and (b) RIT-18 dataset [75] with the Bayesian model uncertainty-based objective estimated by Monte Carlo dropout and computing informative prior maps before each mission starts. Results are averaged over three different UAV starting positions. Shaded regions indicate one standard deviation. Our map-based planners outperform the coverage baseline (yellow) and local planner (purple) with less training data while showing less sensitivity to the UAV starting position.

It verifies our framework's consistently superior active learning performance under (i) varying UAV starting positions, (ii) different pre-training schemes and (iii) different model architectures compared to pre-programmed data collection and state-of-the-art local planning. The experiments are evaluated on the ISPRS Potsdam [68] and RIT-18 [75] datasets using the Bayesian model uncertainty-based objective estimated by Monte Carlo dropout. If not stated otherwise, we use the Bayesian ERFNet pre-trained on Cityscapes [31].

Fig. 6.22 summarises the active learning performance for each planner averaged over three different starting positions at the top-left, top-right, and bottom-right corners of the terrains. All of our map-based planners reach an, on average, higher performance than the baselines (yellow) and local planner (purple) on both datasets. Particularly, the local planner does not perform better on average than the coverage baseline on ISPRS Potsdam. Furthermore, as indicated by the large standard deviations of the local planner and random walk (yellow) on RIT-18, the local planner's and random walk's performances heavily depend on the UAV starting position in challenging to explore terrains. In contrast, our map-based planners are robust to varying UAV starting positions, resulting in superior active learning performance compared to local planners and baselines.

Fig. 6.23 summarises the active learning performance for each planner averaged over three differently pre-trained Bayesian ERFNets. Each mission starts from the top-left terrain corner with Bayesian ERFNet being randomly initialised or pre-trained on Cityscapes [31] or Flightmare [157]. The standard deviations are mainly a result of the randomly initialised models having, as expected, weaker prediction performance than the pre-trained models, irrespective of the planning



(a) ISPRS Potsdam dataset (b) RIT-18 dataset

Figure 6.23: Comparison of active learning performance on the (a) ISPRS Potsdam dataset [68] and (b) RIT-18 dataset [75] with the Bayesian model uncertainty-based objective estimated by Monte Carlo dropout and computing informative prior maps before each mission starts. Results are averaged over three differently pre-trained Bayesian ERFNets. Shaded regions indicate one standard deviation. Our map-based planners outperform the coverage baseline (yellow) and local planner (purple) on average with less training data, irrespective of the pre-training scheme.

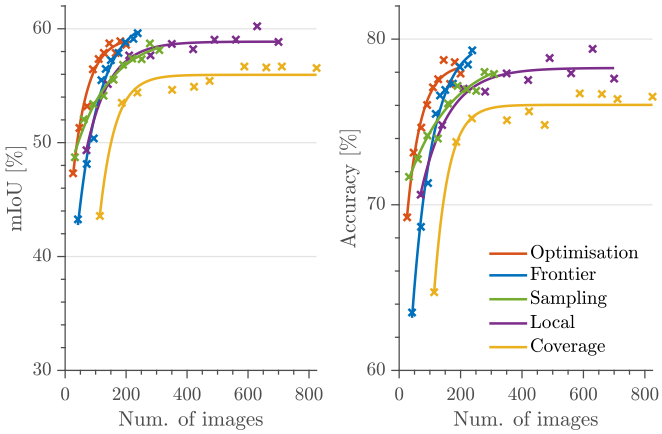


Figure 6.24: Comparison of active learning performance on ISPRS Potsdam [68] using a Bayesian version of U-Net [132] pre-trained on Flightmare [157]. The Bayesian model uncertainty-based planning objective is estimated by Monte Carlo dropout, and informative prior maps are computed before each mission starts. All adaptive planners outperform the coverage baseline (yellow). Our map-based optimisation (orange) and frontier (blue) planners outperform local planning (purple) with less training data.

approach. All our map-based planners show stronger active learning performance on average compared to the baseline approaches (yellow) and the local planner (purple) on both datasets. The local planner fails particularly outperform the random walk (yellow) on RIT-18, irrespective of the pre-training scheme. These findings validate our map-based planners' robustness to varying pre-training schemes.

Fig. 6.24 summarises the active learning performance of our planning framework using a Bayesian variant of U-Net [132]. We extend the U-Net architecture by adding dropout layers after each convolutional block with a dropout probabil-

ity of 10%. Like this, we perform Monte Carlo dropout at inference to compute the uncertainty-based planning objective in Eq. (6.3). We conduct experiments with the Bayesian U-Net pre-trained on Flightmare [157] using the ISPRS Potsdam dataset, starting each mission from the top-left corner. All adaptive planners exceed the maximum prediction performance of the coverage baseline (yellow) with less than half of the training images. This confirms the effectiveness of adaptive planning for active learning, irrespective of the model architecture. Furthermore, our map-based frontier (blue) and optimisation (orange) planners outperform local planning (purple), while the sampling planner (green) performs on par with local planning. This showcases the strong performance of our map-based planners and our framework's applicability to different model architectures.

6.3 Conclusion

Adaptive informative path planning methods, including the ones presented in previous Chap. 4 and Chap. 5, achieve strong performance in robotic information gathering as they allow the robot to perform online decision-making. In this way, robots are equipped with a greater level of autonomy and act more efficiently when deployed in unknown environments. However, adaptive IPP methods assume robust onboard robotic vision to interpret sensor measurements semantically. Modern robotic vision systems rely on deep learning-based semantic segmentation models trained on static human-labelled datasets. As sensor measurements in unknown environments often deviate from the ones the vision model was trained on, semantic segmentation performance in the environment the robot is deployed in commonly degrades. Overall, this results in degraded robotic information-gathering efficiency, requiring costly pixel-wise human labelling of collected images to improve the vision model upon re-training.

To address this issue, in this chapter, we investigated the thesis' second research question of how to improve a robot's deep learning-based vision model in unknown environments with a minimal number of human-labelled training images. Our main contribution is a novel adaptive IPP framework for active learning in semantic terrain mapping. A key aspect of our method is linking the planning objective to active learning acquisition functions. This allows us to adaptively replan the robot's paths towards areas of potentially informative new training data. To ensure maximally informed online decision-making, our global planning algorithms leverage a sequentially updated probabilistic terrain map capturing semantics and acquisition function information. The framework provides diverse acquisition functions, proposes various map-based planning algorithms and is agnostic to the model architecture. The experimental results show that our map-based adaptive planning methods reduces the number of human-labelled

images required to maximise semantic segmentation performance compared to pre-programmed data collection campaigns and state-of-the-art local planning for active learning [12]. Furthermore, our probabilistic global mapping of gathered information enhances map-based planning performance for active learning. Finally, our Bayesian extension of the deterministic ERFNet [131] improves semantic segmentation performance and yields more consistent model uncertainty estimates, resulting in higher planning performance for active learning than previously used non-Bayesian planning objectives.

Our results demonstrate that, in response to the second research question posed in this thesis, integrating adaptive IPP methods with active learning acquisition functions from computer vision reduces the human labelling effort while improving deep learning-based semantic vision models. However, our framework relies on pixel-wise human-labelled images for model re-training, still inducing substantial human labelling costs. Thus, in Chap. 7, we propose a novel semi-supervised learning method to further reduce the human labelling effort. We also show how to integrate this semi-supervised learning approach into our adaptive planning framework for active learning introduced in this chapter.

Chapter 7

Semi-Supervised Learning of Semantic Vision Using Adaptive Informative Path Planning

Perceiving and understanding complex environments is a crucial prerequisite for deploying autonomous robots in diverse environments. Robotic monitoring missions require the robot to perceive and reason about environments beyond geometry, e.g. semantically segmenting collected RGB images. Thus, robotic vision systems need to reliably semantically interpret onboard sensor measurements in novel domains and previously unseen environments. At the same time, the robot is often resource-constrained, which requires the robot to explore the environment efficiently, e.g. within a limited flight time. Recent advances in adaptive IPP, including the methods presented in Chap. 4 and Chap. 5, enable resource-constrained robots to explore unknown environments more efficiently. However, classical deep learning-based semantic vision systems are trained on static datasets. These static datasets do not completely cover the various domains and semantics encountered during real-world deployments in diverse unknown environments. This often results in degraded semantic segmentation performance during deployment.

To this end, we have presented an adaptive IPP framework for active learning in semantic terrain monitoring in Chap. 6. The framework aims to maximise the robot's semantic segmentation performance while minimising the number of human-labelled images. Although our approach shows promising performance, it relies on the full supervision of a human annotator, providing costly pixel-wise semantic annotations for each newly collected training image. To facilitate robot deployment in unknown environments and achieve higher levels of autonomy in robotic information gathering, robotic active learning methods need to request easier-to-annotate and fewer labelling queries to reduce human labelling efforts.

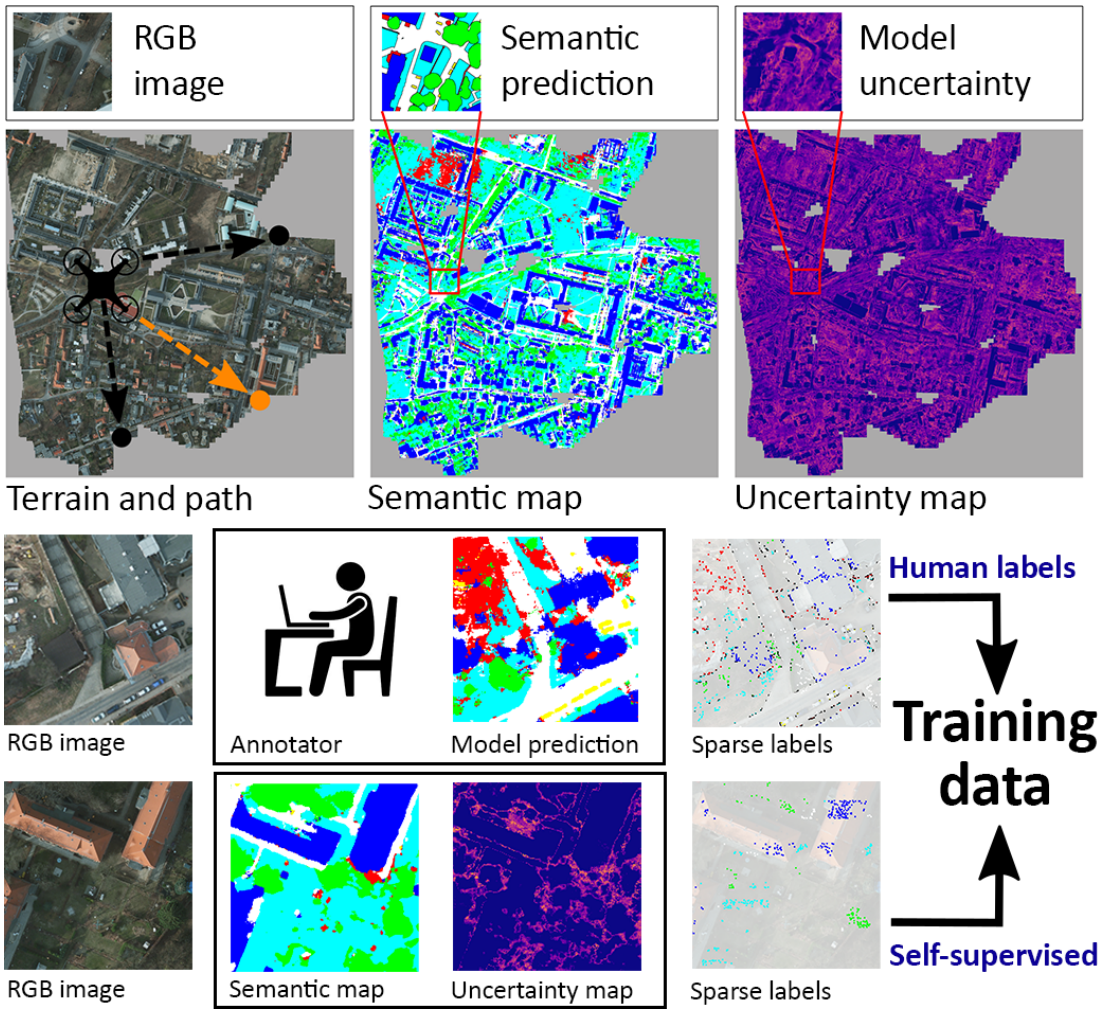


Figure 7.1: Our semi-supervised active learning approach in an unknown terrain (top). We infer semantics (top-centre) and model uncertainty (top-right) and fuse both in terrain maps. The robot adaptively re-plans its path online (orange, top-left) to collect diverse uncertain (yellower, top-right) images. After each mission, we select sparse sets of pixels for human and self-supervised labelling (bottom). Self-supervised labels are rendered from low-uncertainty (blue) semantic map areas. Human labels are queried for cluttered model prediction areas.

Motivated by recent advances in semi-supervised learning [25, 60] and efficient labelling paradigms [149, 181], this chapter examines the problem of semi-supervised active learning to improve robotic semantic vision in unknown terrains. We aim to maximise the robot's semantic segmentation performance while minimising human labelling requirements. The robot adaptively re-plans paths online to collect informative training data to re-train a semantic segmentation model after a mission. We incorporate two sources of labels for model re-training based on the collected data: (i) a human annotator and (ii) automatically generated pseudo labels based on a semantic terrain map incrementally built online.

In the traditional setting, active learning methods select the most informative images from a large unlabelled dataset [44, 47, 147, 183] to reduce human labelling

effort. However, these approaches are typically not applicable to robot deployments in unknown environments, as the collected data is unknown before deployment. Recent approaches combine self-supervised active learning with planning to improve the robot's semantic vision in unknown environments [20, 191]. Self-supervised methods automatically generate pseudo labels from semantic maps incrementally built during a mission [20, 45, 191]. These approaches do not rely on human labels. However, their applicability to unknown environments is often limited since they require large labelled in-domain pre-training datasets to produce high-quality pseudo labels without systematic prediction errors [20].

The main contribution of this chapter is a novel semi-supervised adaptive IPP approach for robotic active learning of semantic segmentation in unknown terrains illustrated in Fig. 7.1. Our approach bridges the gap between the general applicability of fully supervised robotic active learning methods and the low human labelling requirements of self-supervised robotic active learning methods. A key novelty of our semi-supervised robotic active learning method is combining the selection of sparse and informative human-labelled training data and automatically generating uncertainty-aware pseudo labels. We fuse semantic model predictions and Bayesian model uncertainty estimates [76] into terrain maps. Based on the model uncertainty map, our planner adaptively collects images from terrain areas with high model uncertainty and training data diversity. Inspired by recent semi-supervised computer vision works [149, 181], we select only a sparse set of to-be-human-labelled informative pixels from each image using a novel selection criterion. To further improve model performance, we automatically render pseudo labels from the semantic terrain map in areas of low model uncertainty. By combining human and pseudo labels, we aim to maximise semantic segmentation performance while reducing human labelling effort.

In sum, we make three key claims in this chapter. First, our semi-supervised approach drastically reduces the number of human-labelled pixels compared to fully supervised robotic active learning approaches. We preserve similar semantic segmentation performance and outperform self-supervised methods. Second, selecting sparse human labels in our targeted way improves semantic segmentation performance while minimising overall human labelling efforts. Third, the uncertainty-aware generation of pseudo labels further improves semantic segmentation performance compared to learning from sparse human labels alone.

This chapter incorporates material from the following peer-reviewed journal publication, for which I have been the main contributor:

- Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Semi-Supervised Active Learning for Semantic Segmentation in Unknown Environments Using Informative Path Planning. *IEEE Robotics and Automation Letters (RA-L)*, 9(3):2662–2669, 2024

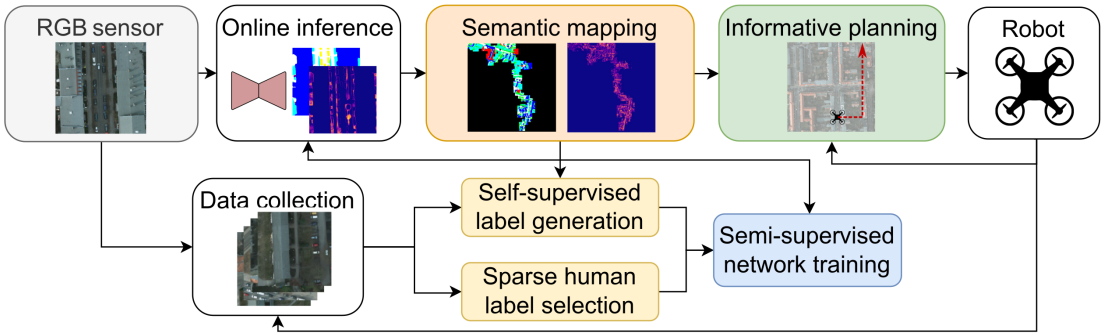


Figure 7.2: During a mission, a semantic segmentation model predicts pixel-wise semantics and model uncertainties from an RGB image. Both are fused into an uncertainty-aware semantic terrain map. Our frontier planner guides the collection of training data for model re-training based on the robot state and map belief towards areas of high model uncertainty and training data diversity. After a mission, the collected data is labelled using two sources of labels: (i) a human annotator labels a sparse set of informative pixels, and (ii) we automatically render pseudo labels from the semantic map in an uncertainty-aware fashion.

7.1 Adaptive Informative Path Planning Algorithm

We present a semi-supervised robotic active learning approach for semantic segmentation using adaptive IPP as illustrated in Fig. 7.2. We collect images in an unknown terrain using a robot with an RGB sensor to improve semantic vision with minimal human labelling effort. We predict pixel-wise semantics and model uncertainties to update a probabilistic semantic terrain map, described in Sec. 7.1.1. Based on the robot's position, budget, and map belief, we adaptively re-plan paths to collect training data in areas of high model uncertainty and training data diversity, explained in Sec. 7.1.2. After a mission has been completed, we select a sparse set of informative to-be-human-labelled pixels in the collected images, detailed in Sec. 7.2.1. Additionally, we automatically render pseudo labels from the online-built semantic map in areas of low model uncertainty, outlined in Sec. 7.2.2. We combine both sources of labels for model re-training.

7.1.1 Probabilistic Semantic Environment Mapping

A crucial requirement for pseudo label generation and adaptive planning is a probabilistic map capturing information about the terrain. We use a probabilistic multi-layered semantic terrain mapping to fuse semantic information. The terrain is discretised into two grid maps $\mathcal{M}_S : G \rightarrow \{0,1\}^{K\times W\times L}$ and $\mathcal{M}_U : G \rightarrow [0,1]^{W\times L}$ defined over $W \times L$ spatially independent grid cells G. The map $\mathcal{M}_S$ captures the semantic model predictions, and the map $\mathcal{M}_U$ stores the model uncertainties associated with the mapped semantic model predictions $\mathcal{M}_S$.

The semantic map $\mathcal{M}_S$ consists of K layers with one layer per semantic class. At each time step, a new RGB image arrives. The probabilistic semantic model predictions and model uncertainties are inferred using a semantic segmentation model and Monte Carlo dropout [46]. Specifically, we use the Bayesian ERFNet proposed in Sec. 6.1.1. We project the semantic model predictions and model uncertainties on the terrain map using the robot's position and the camera's field of view. Each layer in the semantic map $\mathcal{M}_S$ is recursively updated using occupancy grid mapping [114]. The model uncertainty map $\mathcal{M}_U$ is updated by maximum likelihood estimation. Our approach is agnostic to the chosen model architecture and uncertainty estimation technique. Furthermore, it can be integrated with other model architectures and uncertainty estimation techniques.

Additionally, we maintain a count map $\mathcal{M}_T : G \to \mathbb{N}^{W \times L}$ to track the occurrences in the human-labelled training data and a hit map $\mathcal{M}_H : G \to \mathbb{N}^{W \times L}$ to update estimates of model uncertainties $\mathcal{M}_U$. The maps $\mathcal{M}_T$ and $\mathcal{M}_H$ are not identical as the camera could provide a high-frequency image stream for mapping semantics, uncertainties and hits. In contrast, only images at the planned measurement position are collected for human-labelled training data. Both maps are used during adaptive replanning of the robot's path. We detect frontiers of unexplored space in the hit map and quantify training data diversity in our planning objective based on the training data count map.

The semantic predictions and model uncertainties change as the semantic segmentation model is re-trained after each robot mission. Following our informative mapping module in Sec. 6.1.2, we re-compute the semantic and model uncertainty maps after model re-training using all previously collected RGB images. In this way, we obtain maximally up-to-date prior maps for adaptive planning.

7.1.2 Frontier-Based Planning for Active Learning

Our planner is designed to collect new training data in unknown terrains, taking into account mission budget constraints. We aim to maximise the performance of a semantic segmentation model with minimal human labelling effort after having re-trained it on the collected training data. Our planning method searches for a path $\psi^* = (\mathbf{p}_1, \dots, \mathbf{p}_N) \in \Psi$ with $N \in \mathbb{N}$ robot positions $\mathbf{p}_i \in \mathbb{R}^3$, $i \in \{1, \dots, N\}$, in the set of potential paths Ψ, that maximises an information criterion $I : \Psi \to \mathbb{R}$ while respecting the limited mission budget $B \geq 0$ given a function $C : \Psi \to \mathbb{R}$ assigning the cost $C(\psi)$ of executing a path ψ.

At each time step t, we adaptively re-plan the next-best robot position $\mathbf{p}^*_{t+1}$ to collect informative training data based on the current model uncertainty map $\mathcal{M}^t_U$, human-labelled training data count map $\mathcal{M}^t_T$, and hit map $\mathcal{M}^t_H$. We use a geometric frontier-based planner [182] guided by the information criterion I. The information criterion estimates the effect of a training image recorded at a can-

didate robot position on a semantic segmentation model's performance. Based on the hit map $\mathcal{M}_H^t$, we assign each grid cell $g \in G$ to one of the disjoint sets $G_U \cup G_O = G$ containing the unknown and known gird cells, respectively. A grid cell $g \in G_O$ is considered to be known if its hit count is $\mathcal{M}_H(g) > 0$. A grid cell $g \in G_U$ is considered to be unknown if its hit count is $\mathcal{M}_H(g) = 0$. Then, a frontier of explored space is defined as a connected set of known grid cells $g \in G_O$, where each known cell has a neighbouring unknown grid cell in G_U [182].

We generate a set of potentially informative next robot position candidates $\mathcal{P}_{t+1}^c$, which are equidistantly sampled positions $\mathbf{p}_{t+1}^c \in \mathcal{P}_{t+1}^c$ along the frontiers of known and unknown space reachable within the remaining mission budget $B_t \leq B$ at a fixed altitude. At each time step t, the planner greedily selects the candidate frontier position $\mathbf{p}_{t+1}^c \in \mathcal{P}_{t+1}^*$ with the highest information value,

$$\mathbf{p}_{t+1}^* = \underset{\mathbf{p}_{t+1}^c \in \mathcal{P}_{t+1}^c}{\operatorname{argmax}} \; I(\mathbf{p}_{t+1}^c) = \underset{\mathbf{p}_{t+1}^c \in \mathcal{P}_{t+1}^c}{\operatorname{argmax}} \; \sum_{g \in \mathrm{Img}(\mathbf{p}_{t+1}^c)} \begin{cases} c_u & \text{, if } g \in G_U \\ \frac{\mathcal{M}_U^t(g)}{\mathcal{M}_T^t(g)} & \text{, if } g \in G_O \,, \end{cases} \tag{7.1}$$

where $c_u \in \mathbb{R} \geq 0$ is a constant model uncertainty prior that fosters exploration of unobserved areas. The set $\mathrm{Img}(\mathbf{p}_{t+1}^c)$ contains grid cells visible from position $\mathbf{p}_{t+1}^c$, obtained by projecting the camera's field of view at position $\mathbf{p}_{t+1}^c$ on the terrain. As model uncertainty is found to be reducible by adding more training data about which the model is uncertain [76], it possibly increases prediction performance. Thus, an already observed grid cell's g effect on semantic segmentation performance upon model re-training is estimated to be high if its model uncertainty $\mathcal{M}_U^t(g)$ is also high. To balance between model uncertainty and human-labelled training data diversity, we normalise a grid cell's information value by its number of occurrences in the human-labelled training dataset $\mathcal{M}_T^t(g)$.

7.2 Semi-Supervised Model Training

The main contribution of our approach is a semi-supervised model training strategy for improving the robot's semantic vision. We use a semantic segmentation model $f^{\mathbf{w}}$ parameterised by weights $\mathbf{w}$ to predict the pixel-wise probabilities $p(\cdot \,|\, \mathbf{z}, \mathbf{w}) = \mathrm{softmax}(\mathbf{f}^{\mathbf{w}}(\mathbf{z})) \in [0,1]^{K \times w \times h}$ over all possible semantic labels $\mathbf{y} \in \{1, \ldots, K\}^{w \times h}$ given image $\mathbf{z}$ of resolution $w \times h$. We follow Sec. 6.1.1.1 to estimate pixel-wise model uncertainties $\mathbf{u} \in [0,1]^{w \times h}$ via Monte Carlo dropout [46].

To maximise semantic segmentation prediction performance, we combine human labels $\mathcal{Y}_l = \{\mathbf{y}_l^1, \ldots, \mathbf{y}_l^{N_l}\}$ of images $\mathcal{Z}_l = \{\mathbf{z}_l^1, \ldots, \mathbf{z}_l^{N_l}\}$ with pseudo labels $\mathcal{Y}_u = \{\mathbf{y}_u^1, \ldots, \mathbf{y}_u^{N_u}\}$ of images $\mathcal{Z}_u = \{\mathbf{z}_u^1, \ldots, \mathbf{z}_u^{N_u}\}$, where N_l and N_u are the numbers of human-labelled and pseudo-labelled images. To reduce human labelling effort, we select only a sparse set of to-be-human-labelled pixels from each image $\mathbf{z}_l^i$, where $i \in \{1, \ldots, N_l\}$, as described in Sec. 7.2.1. To balance human

and self-supervision during model training, we also select only a sparse set of pseudo-labelled pixels from each image $\mathbf{z}_u^i$, where $i \in \{1, \ldots, N_u\}$ as described in Sec. 7.2.2. Each non-labelled pixel in some label $\mathbf{y}_l^i$ or $\mathbf{y}_u^i$ is assigned a void class N^v. During training, we mask the loss with $\mathbb{I}_{\mathbf{y} \neq N^v} \in \{0,1\}^{w \times h}$, where $\mathbb{I}_{\mathbf{y} \neq N^v}$ is zero for each pixel with class N^v. The model $f^{\mathbf{w}}$ is trained to minimise the cross-entropy loss function with weight decay coefficient λ,

$$\begin{aligned} \mathcal{L}(\mathbf{w}) = & \frac{1}{N_l \alpha} \sum_{i=1}^{N_l} \left\| -\log\left(p\left(\mathbf{y}_l^i \,|\, \mathbf{z}_l^i, \mathbf{w}\right)\right) \odot \mathbb{I}_{\mathbf{y}_l^i \neq N^v} \right\|_1 \\ & + \frac{1}{N_u \alpha} \sum_{i=1}^{N_u} \left\| -\log\left(p\left(\mathbf{y}_u^i \,|\, \mathbf{z}_u^i, \mathbf{w}\right)\right) \odot \mathbb{I}_{\mathbf{y}_u^i \neq N^v} \right\|_1 + \lambda \|\mathbf{w}\|_2^2, \end{aligned} \tag{7.2}$$

where $\alpha \in \mathbb{N}$ is the number of labelled pixels per image, the log-operator is applied element-wise to the probability matrix, $\|\cdot\|_1$ sums the elements of a matrix, $\odot$ is the Hadamard product performing element-wise multiplication of two matrices, and $\|\cdot\|_2$ is the Euclidean norm of a vector.

7.2.1 Sparse Human Labelling Query Selection

Inspired by Shin et al. [149] and Xie et al. [181], we propose a new model architecture-agnostic pixel selection procedure for sparse human labels. Our pixel selection method balances label informativeness and training data diversity. After each mission, for all newly collected images $\mathbf{z}_l$ recorded at planned positions $\mathbf{p}_t^*$ according to Eq. (7.1), for each pixel (m, n), we predict semantic probabilities $p(\cdot \,|\, \mathbf{z}_l, \mathbf{w})^{(m,n)} \in [0,1]^K$. Based on this probabilistic prediction, we extract the maximum likelihood label $\tilde{\mathbf{y}}_l^{(m,n)} = \operatorname{argmax}_{k \in \{1,\ldots,K\}} p(k \,|\, \mathbf{z}_l, \mathbf{w})^{(m,n)}$. Next, for each newly collected image $\mathbf{z}_l$, based on its prediction $\tilde{\mathbf{y}}_l$, we compute each pixel's region impurity score following Xie et al. [181] as

$$\begin{aligned} R_r(\mathbf{z}_l)^{(m,n)} &= -\sum_{k=1}^{K} \log\left(\frac{|N_r^k(m,n)|}{(2r+1)^2}\right) \frac{|N_r^k(m,n)|}{(2r+1)^2}, \\ N_r^k(m,n) &= \left\{(i,j) \in N_r(m,n) \,|\, \tilde{\mathbf{y}}_l^{(i,j)} = k\right\}, \end{aligned} \tag{7.3}$$

where $N_r(m,n) = \{(i,j) \mid |i-m| \leq r \wedge |j-n| \leq r\}$ is the set of r-step neighbouring pixels of pixel (m,n), and $|\cdot|$ counts the elements in a set. The region impurity of a pixel is high whenever the number of different classes predicted within the pixel's r-step neighbourhood is high. Hence, high region impurity indicates locally cluttered predictions. However, a well-trained model typically predicts locally non-cluttered semantics. Thus, high region impurity potentially indicates high human label information value upon model re-training.

In contrast to Xie et al. [181], we do not greedily select the α image pixels that maximise region impurity. Instead, we sample α image pixels uniformly at

random from the $\beta\,\%$ pixels with the highest region impurity score per image. Through this, we foster human label diversity. The user can set α dynamically to steer the desired human labelling effort. At the same time, β implicitly provides a lower bound for a pixel's information value. This lower bound aims to ensure that human-labelled pixels are informative and maximise model improvements. Experimentally, we found that values $\beta \leq 10\,\%$ ensure informative pixel selection. Setting $\beta \to 100\,\%$ results in random pixel selection. Particularly for desirably small labelling budgets α, random pixel selection fails to effectively improve model performance. Furthermore, we found that greedily selecting the α pixels with the highest region impurity tends to select similar training data, resulting in worse prediction performance. Thus, both region impurity and random sampling are crucial for maximising model performance with low human labelling budgets.

7.2.2 Self-Supervised Pseudo Label Generation

After a mission is finished, we use our incrementally online-built uncertainty-aware semantic map introduced in Sec. 7.1.1 to generate pseudo labels $\mathcal{Y}_u$ in a self-supervised fashion, similar to self-supervised approaches [20, 45, 191]. We record to-be-pseudo-labelled images $\mathbf{z}_u \in \mathcal{Z}_u$ equidistantly between two positions planned for collecting to-be-human-labelled images, see Eq. (7.1). In this way, we aim to maximise the diversity of our pseudo-labelled training data. Given a robot position $\mathbf{p}_u$ at which image $\mathbf{z}_u$ was recorded, we render a pixel-wise probabilistic pseudo label $p(\cdot \,|\, \mathbf{p}_u, \mathcal{M}_S) \in [0,1]^{K \times w \times h}$ from the semantic map $\mathcal{M}_S$ at the image resolution $w \times h$. To render the map-based semantics, we project the camera's field of view at position $\mathbf{p}_u$ on the terrain. Then, for each pixel (m,n) of $\mathbf{z}_u$, we extract the maximum likelihood pseudo label $\mathbf{y}_u^{(m,n)} = \operatorname{argmax}_{k \in \{1,\dots,K\}} p(k \,|\, \mathbf{p}_u, \mathcal{M}_S)^{(m,n)}$. In addition, we render the pseudo label's corresponding pixel-wise model uncertainty $\mathbf{u}_u \in [0,1]^{w \times h}$ from the model uncertainty map $\mathcal{M}_U$.

In contrast to previous self-supervised methods [20, 45, 191], we only use a sparse set of α pseudo-labelled pixels per image $\mathbf{z}_u$ to train the model via Eq. (7.2). We experimentally found sparse pseudo labels to better balance between human and self-supervision. We extend the approach of Shin et al. [149] to a new pixel selection procedure for sparse pseudo labels $\mathbf{y}_u$. Our selection procedure is a trade-off between semantic map uncertainty and pseudo label diversity. After each mission, we (re-)render pseudo labels $\mathbf{y}_u$ and model uncertainties $\mathbf{u}_u$ based on the most recent map beliefs $\mathcal{M}_S$ and $\mathcal{M}_U$ for all images $\mathbf{z}_u$ collected in any of the previous missions. Similar to the human-labelled pixel selection in Sec. 7.2.1, for each image, we sample α pixels (m,n) at random from the $\beta\%$ pixels with the lowest map-based model uncertainty $\mathbf{u}_u^{(m,n)}$. We found that providing an implicit upper bound β for model uncertainty yields higher semantic segmentation performance than random sampling pseudo-labelled pixels as $\beta \to 100\,\%$. The

upper bound on model uncertainty for pseudo-labelled pixels acts as a proxy for an implicit lower bound on self-supervision quality. Experimentally, we found that $\beta \leq 10\,\%$ usually ensures moderate model performance improvements.

Our semi-supervised training can easily be expanded by independently tuning different values of α and β for the pixel selection of human and pseudo labels. We, however, prioritise simplicity and reduce the number of user-defined parameters to a minimum. Thus, we deploy our approach using the same number of selected pixels α and bounds β for human- and pseudo-labelled images.

7.3 Experimental Evaluation

Our experiments assess the performance of our semi-supervised robotic active learning approach and investigate our claims made in this chapter. In Sec. 7.3.2, we show that our method for selecting human-labelled pixels outperforms state-of-the-art pixel selection methods in our robotic planning context. In Sec. 7.3.3, we validate that combining our uncertainty-aware pseudo labels with human labels improves semantic segmentation performance. Furthermore, we verify that our semi-supervised method drastically reduces the number of human-labelled pixels compared to fully supervised robotic active learning approaches while maintaining similar performance. Finally, in Sec. 7.3.4, we show that our semi-supervised approach outperforms self-supervised methods in semantic terrain mapping.

7.3.1 Experimental Setup

Baseline & Dataset. We compare our semi-supervised frontier-based planning approach against a coverage-based strategy that pre-computes paths to maximise spatial coverage. Our approach is evaluated on the real-world 7-class urban ISPRS Potsdam orthomosaic dataset [68] by simulating 10 subsequent UAV missions from 30 m altitude with a mission budget of 1800 s. The UAV uses a downwards-facing RGB camera with a footprint of $400\,\text{px} \times 400\,\text{px}$, resulting in 15 cm ground sampling distance, identical in setup with Sec. 6.2 to ensure a fair comparison with our fully supervised robotic active learning approach presented in Chap. 6.

Evaluation Metrics. We evaluate semantic segmentation performance (dependent variable) over the number of human-labelled training images or pixels (independent variable). We use mean Intersection-over-Union (mIoU) [31] and pixel-wise accuracy [33] to quantify semantic segmentation performance. Higher semantic segmentation performance thanks to newly added images indicates better robotic active learning performance. We run three trials per experiment to account for inherent randomness in the pixel selection and model training. We report the mean and standard deviation performance curves.

Implementation Details. We use Bayesian ERFNet as in Sec. 6.1.1, which has been pre-trained on the Cityscapes dataset [31]. For a fair comparison, re-training after each mission starts from this checkpoint and the model is trained until convergence on a validation set. We use a one-cycle learning rate, a batch size of 8, and weight decay $\lambda = (1 - p)/2N$ in Eq. (7.2), where $p = 0.5$ is the dropout probability, and $N = N_l + N_u$ is the number of human- and pseudo-labelled training images [47]. The human and pseudo label pixel selection bounds are set to $\beta = 5\%$. The r-neighborhood of the human label selection criterion is set to $r = 1$ in Eq. (7.3) as we found considering small regions to perform best.

7.3.2 Targeted Human Label Selection

The first set of experiments shows that our targeted human label selection improves semantic segmentation performance and reduces human labelling effort, supporting our second claim made in this chapter. We verify that our method (i) outperforms state-of-the-art pixel selection methods in the robotic active learning context and (ii) improves semantic segmentation performance over non-targeted pixel selection with desirably higher gains for lower human labelling budgets. The experiments are conducted using human labels only.

We compare our human-labelled pixel selection method introduced in Sec. 7.2.1, referred to as *Ours*, against four other pixel selection methods. Namely: (i) sample α pixels from the $\beta\%$ most uncertain pixels by Shin et al. [149], referred to as *Unc-Rand*; (ii) sample $\beta\%$ pixels at random, then select the α most uncertain pixels by Shin et al. [149], referred to as *Rand-Unc*; (iii) select α pixels uniformly at random, referred to as *Random*; and (iv) select α pixels with the highest region impurity in an r-neighborhood by Xie et al. [181], referred to as *Reg-Imp*, where $r = 1$ yields the best results. We set a low human labelling budget of $\alpha = 1000 \approx 0.6\%$ pixels for each collected image by our frontier planner. Additionally, we show results for the fully supervised frontier-based and coverage-based planners, referred to as *Frontier* and *Coverage* respectively, using pixel-wise densely human-labelled images investigated in Sec. 6.2 as a performance upper bound for the sparsely labelled approaches.

Fig. 7.3 summarises the semantic segmentation performance of the different pixel selection methods. In line with our fully supervised robotic active learning approach presented in Chap. 6, the frontier planner (yellow) using densely human-labelled images achieves the highest performance outperforming the non-adaptive coverage planner (orange). Notably, our method (dark blue) shows the fastest improvement and highest final mIoU of approx. 52.5% of all pixel selection methods. This verifies that our method outperforms the second-best state-of-the-art Reg-Imp method (green), reaching approx. 49% final mIoU. Particularly, our human label selection matches the final performance of the fully supervised cov-

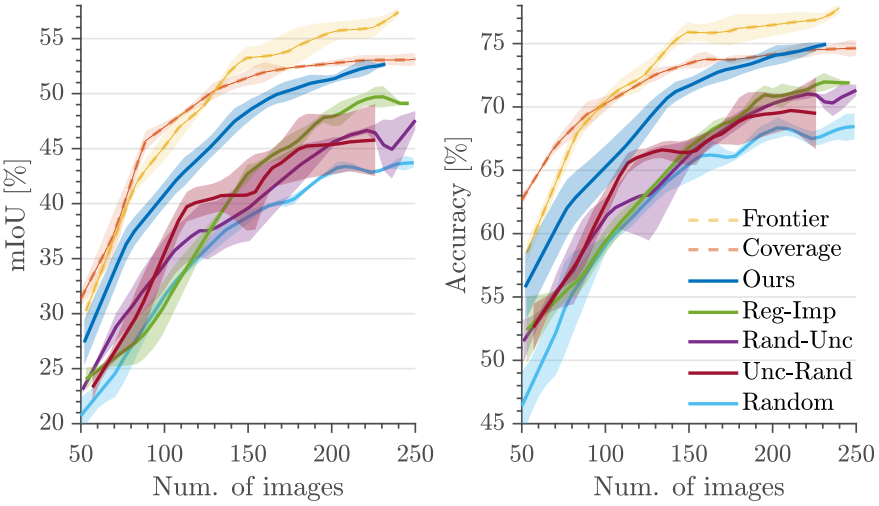


Figure 7.3: Comparison of pixel selection methods with $\alpha = 1000$ human-labelled pixels per image using our frontier planner on ISPRS Potsdam. Frontier (yellow) and coverage (orange) planners use densely labelled images indicating performance upper bounds. Results are averaged over three runs. Shaded regions indicate one standard deviation. Our proposed method (dark blue) outperforms the other state-of-the-art pixel selection methods.

Table 7.1: Per-class IoU comparison of sparse human label selection methods with $\alpha = 1000$ human-labelled pixels per image using our frontier planner on ISPRS Potsdam. 'Dense' uses dense pixel-wise human-labelled images indicating the performance upper bound.

Method	Mission	Surface	Building	Vegetation	Tree	Car	Clutter
Random	3	53.98	51.00	40.58	20.87	28.54	7.58
Unc-Rand		58.93	60.89	43.09	25.15	42.04	11.42
Reg-Imp		51.17	48.84	39.96	15.29	0.00	8.26
Ours		**59.47**	**65.74**	**46.37**	**33.74**	**47.20**	**15.36**
Dense		63.93	70.39	49.46	35.82	60.95	10.40
Random	6	59.16	63.30	43.33	31.62	44.68	11.63
Unc-Rand		61.87	68.99	42.50	29.80	52.57	**16.60**
Reg-Imp		60.19	69.61	46.68	30.83	59.49	12.59
Ours		**65.99**	**72.83**	**51.56**	**41.16**	**61.07**	15.53
Dense		71.08	77.72	53.14	45.80	68.81	17.56
Random	9	59.38	64.71	43.80	33.21	50.54	11.33
Unc-Rand		62.40	70.19	46.68	30.91	57.32	14.92
Reg-Imp		62.31	71.78	46.87	36.92	64.57	12.33
Ours		**67.94**	**74.54**	**52.00**	**43.37**	**66.50**	**16.67**
Dense		71.23	78.60	52.79	48.52	71.57	20.11

erage planner while using only approx. 0.6% of the human-labelled pixels. This highlights the benefits of coupling adaptive planning for collecting training images with our targeted selection of human-labelled pixels for model re-training. Tab. 7.1 shows a per-class IoU performance compared with state-of-the-art pixel

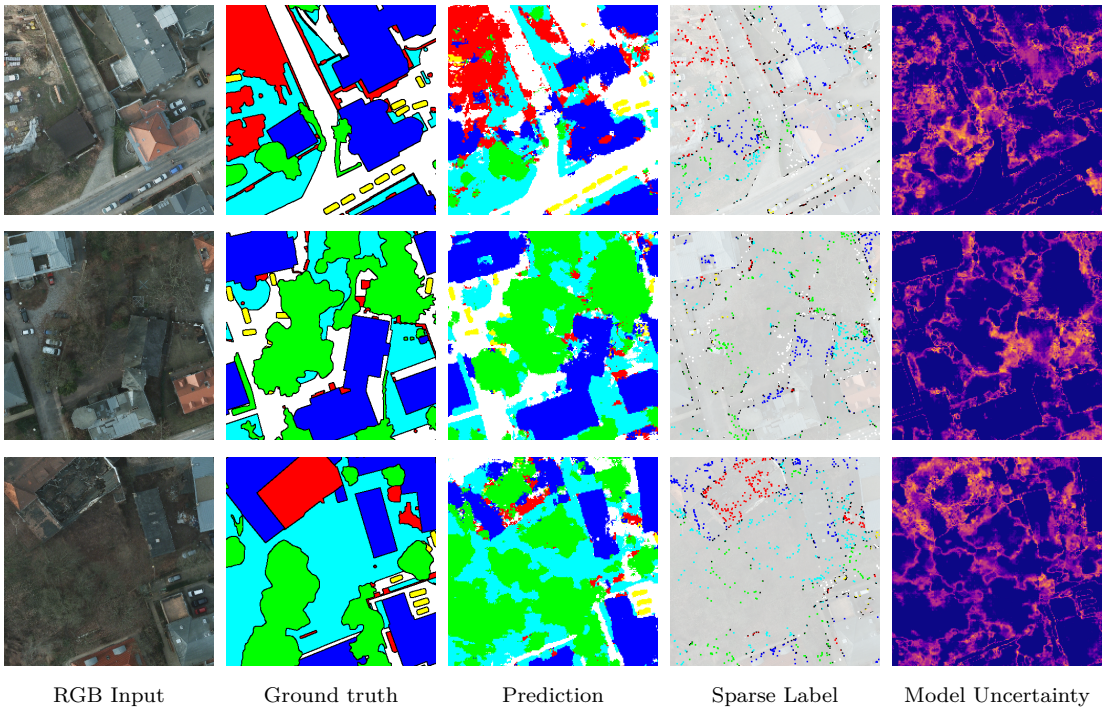


Figure 7.4: Qualitative results of our human label pixel selection method on ISPRS Potsdam. Columns: RGB input, ground truth, prediction, selected to-be-labelled pixels, model uncertainty. Selected pixels are expanded to their one-pixel neighbourhood for visualisation. Our method selects pixels in areas of cluttered predictions, often corresponding to prediction errors.

selection methods. Our method shows superior per-class performance throughout all semantic classes. This verifies our method's ability to select informative pixels for vastly different semantic classes. Fig. 7.4 displays images collected during a mission, semantic predictions and corresponding human-labelled pixels selected for re-training with our method. Favorably, our method selects pixels in areas of cluttered predictions, often corresponding to prediction errors.

Fig. 7.5 shows the semantic segmentation performance of our targeted pixel selection method (solid lines) compared to randomly selecting human-labelled pixels (dashed transparent lines) over varying human labelling budgets. Noticeably, for budgets $\alpha \leq 2000 \approx 1.3\,\%\,\text{px}$, our pixel selection method clearly outperforms random pixel selection. As desired, the performance gain of our targeted pixel selection method over the random selection drastically increases with lower human labelling budgets. These results show that our targeted sparse human label selection is necessary to achieve drastic model performance improvements for reasonably low human labelling requirements. Naive random selection of human-labelled pixels cannot ensure large model performance improvements unless human annotators label substantial amounts of pixels. For an extremely low budget of $\alpha = 100 \approx 0.06\,\%\,\text{px}$, our targeted pixel selection method leads to a substantial final performance gain of approx. 20 % mIoU.

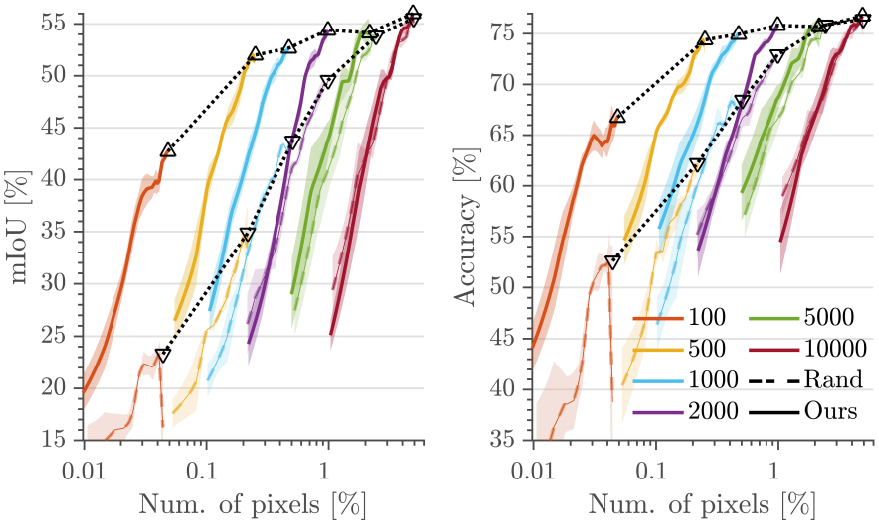


Figure 7.5: Comparison of our human pixel selection method (solid lines) to random pixel selection (dashed transparent lines) over varying labelling budgets $\alpha \in [100, 10000]$ px using our frontier planner on ISPRS Potsdam. Results are averaged over three runs. Shaded regions indicate one standard deviation. As desired, the performance gain of our method compared to random pixel selection drastically increases for lower labelling budgets.

7.3.3 Uncertainty-Aware Pseudo Label Generation

The second set of experiments shows that uncertainty-aware generation of pseudo labels improves semantic segmentation performance, supporting our first and third claims made in this chapter. We validate that our pseudo label selection method outperforms other selection strategies, and combining our human and pseudo label selection improves semantic segmentation performance across varying labelling budgets, supporting our third claim. Furthermore, we verify that our semi-supervised approach drastically reduces the number of human-labelled pixels compared to fully supervised approaches presented in Chap. 6 while maintaining similar performance, supporting our first claim. The experiments are conducted using our human label selection method introduced in Sec. 7.2.1.

We compare our pseudo label selection method introduced in Sec. 7.2.2, referred to as *Ours*, against two other pseudo label selection methods for a low human labelling budget of $\alpha = 1000 \approx 0.6\,\%$ pixels per image. We (i) re-distribute the pseudo labels' class distribution to the true class distribution estimated by the human labels using per-class model uncertainty thresholds to select on average α pixels per image as proposed by He et al. [60], referred to as *Dist-Align*. We also (ii) randomly select α pixels per image, referred to as *Random*. Finally, we compare against (iii) using α human-labelled pixels per image only, referred to as *Human-Only*, and against (iv) using the fully supervised frontier-based and coverage-based planners, referred to as *Frontier* and *Coverage* respectively, leveraging dense pixel-wise human labels as investigated in Sec. 6.2.

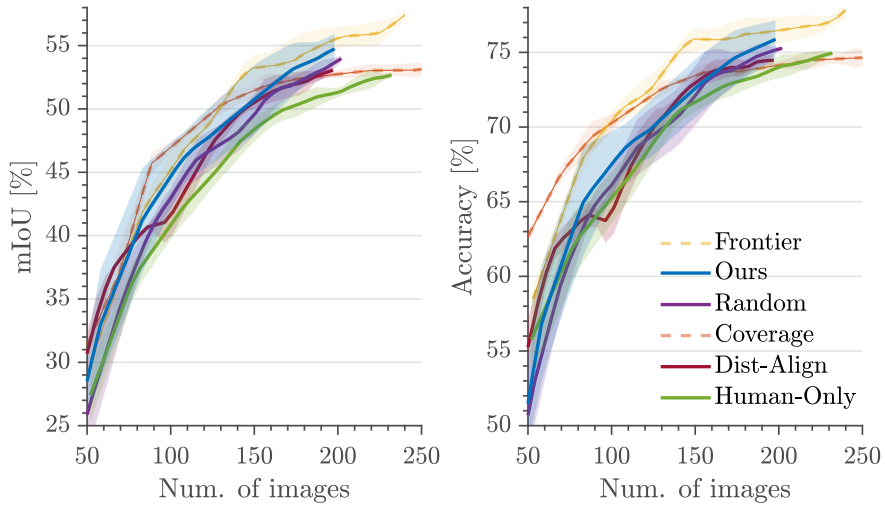


Figure 7.6: Comparison of pseudo label selection methods with $\alpha = 1000$ human- and pseudo-labelled pixels per image using our frontier planner on ISPRS Potsdam. Frontier (yellow) and coverage (orange) planners use densely labelled images indicating performance upper bounds. Results are averaged over three runs. Shaded regions indicate one standard deviation. Our pseudo label selection method (dark blue) outperforms the other selection methods.

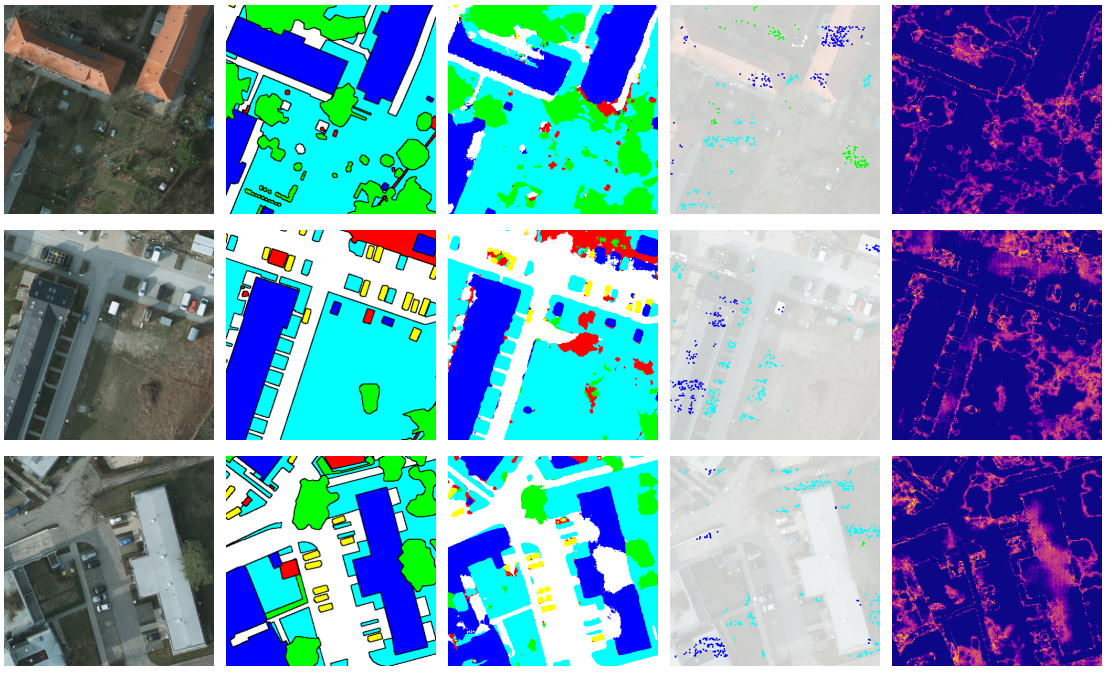


Figure 7.7: Qualitative results of our pseudo label generation on ISPRS Potsdam. Columns from left to right: RGB input, ground truth, pseudo label, selected pseudo-labelled pixels, mapped model uncertainty. Selected pixels are expanded to their one-pixel neighbourhood for visualisation. Our method selects low-uncertainty pixels to minimise pseudo label errors.

Fig. 7.6 summarises the performance of the different pseudo label selection methods. Combining human with pseudo labels improves the semantic segmenta-

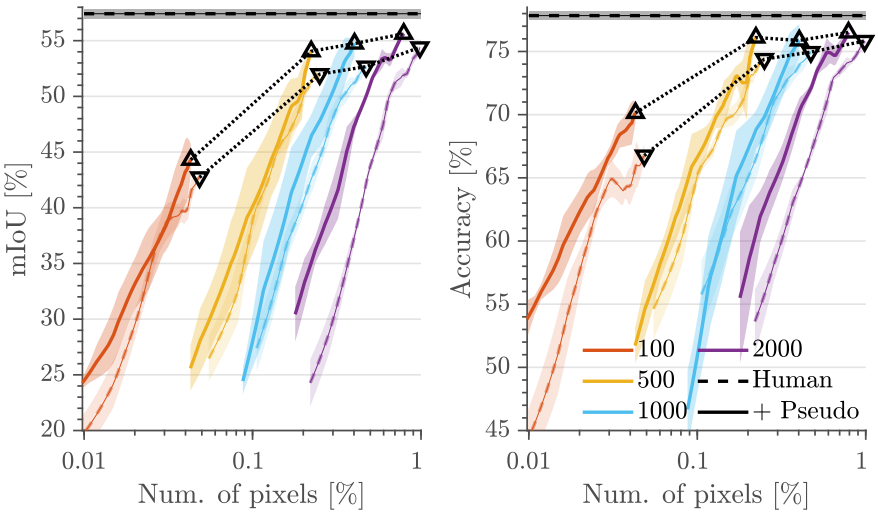


Figure 7.8: Comparison of our human label selection only (dashed transparent lines), and combined with our pseudo label selection (solid lines) over varying labelling budgets $\alpha \in [100, 2000]$ px per image using our frontier planner on ISPRS Potsdam. Dashed black lines show the fully supervised frontier planner's final performance as an upper bound. Results are averaged over three runs. Shaded regions indicate one standard deviation. Combining our pseudo with human labels consistently improves performance over using human labels alone.

tion performance over using sparse human labels alone (green). This verifies the benefits of our semi-supervised robotic active learning approach, combining human and self-supervision. Our uncertainty-aware pseudo label selection method (dark blue) achieves approx. $1-2\%$ higher mIoU than other methods (purple, dark red). Our semi-supervised approach particularly outperforms the fully supervised coverage planner using only approx. $0.6\,\%$ of the human-labelled pixels. Fig. 7.7 qualitatively shows our pseudo labels generated after mission completion.

Fig. 7.8 shows the performance using our human-labelled pixel selection method alone (dashed transparent lines) and combining it with our uncertainty-aware pseudo label selection (solid lines) over varying human labelling budgets $\alpha \in [0.06, 1.25]\,\%$ pixels per image using our frontier planner. Combining our sparse human and pseudo labels consistently improves final semantic segmentation performance by approx. $2-3\,\%$ mIoU across varying human labelling budgets. Furthermore, training on pseudo and human labels simultaneously yields faster performance improvements than using our sparse human labels alone. These results validate the superior performance of our semi-supervised robotic active learning approach. Especially our semi-supervised robotic active learning approach rapidly closes the final performance gap to the fully supervised frontier planning approach proposed in Chap. 6 (dashed black line) with substantially reduced human labelling requirements. The fully supervised approach reaches a maximum performance of approx. 57.5% mIoU while our semi-supervised approach reaches approx. 56% mIoU with only approx. 0.8% of the human-labelled

pixels. This shows that our semi-supervised robotic active learning approach requires two magnitudes fewer human-labelled pixels while reaching semantic segmentation performance similar to fully supervised approaches.

7.3.4 Semi- vs. Self-Supervised Robotic Active Learning

The third set of experiments is designed to support our first claim made in this chapter. We demonstrate that our semi-supervised robotic active learning approach outperforms self-supervised robotic active learning approaches by a large margin regarding semantic segmentation performance under varying human labelling budgets for model pre-training and model re-training in unknown terrains.

Similarly to self-supervised robotic continual learning and domain adaptation approaches [45, 191], we use our frontier planner introduced in Sec. 7.1.2 to guide uncertainty-driven training data collection. We exploit the online-built semantic map to generate densely annotated pseudo labels to establish a self-supervised robotic active learning baseline for semantic terrain monitoring. Current self-supervised robotic active learning approaches only work with pre-trained semantic segmentation models deployed in similar environments [20, 45, 191]. In contrast, our semi-supervised method works in completely unknown terrains. To compare our approach with self-supervised methods, we relax these assumptions. We consider small amounts of densely human-labelled images for pre-training, randomly sampled from the terrain in which the robot is deployed. Each approach starts with this pre-trained model checkpoint. Similar to the experience replay method of self-supervised approaches [45, 191], the human-labelled pre-training images are additionally used for all model re-trainings after a mission is completed to achieve performance improvements in the self-supervised approach.

Fig. 7.9 shows the semantic segmentation performance of our semi-supervised robotic active learning approach (solid lines) compared to the self-supervised robotic active learning approach (dashed lines) on the ISPRS Potsdam dataset with varying numbers of human-labelled pre-training images. For all labelling budgets of $\alpha \in \{100, 500\} \approx \{0.06, 0.3\}\%$ human-labelled pixels per image and $\{16, 32\}$ densely human-labelled pre-training images, our semi-supervised robotic active learning approach outperforms the self-supervised robotic active learning approach by a large margin. With a small number of 16 pre-training images and little human supervision of $\alpha = 100$ pixels per image during the missions, our semi-supervised approach achieves substantially higher final performance than the self-supervised approach with 32 pre-training images. Moreover, the self-supervised approach fails to improve its semantic segmentation performance after five missions, irrespective of the number of pre-training images. This suggests that semi-supervised robotic active learning is necessary for maximally improving semantic segmentation during deployment in varying unknown terrains.

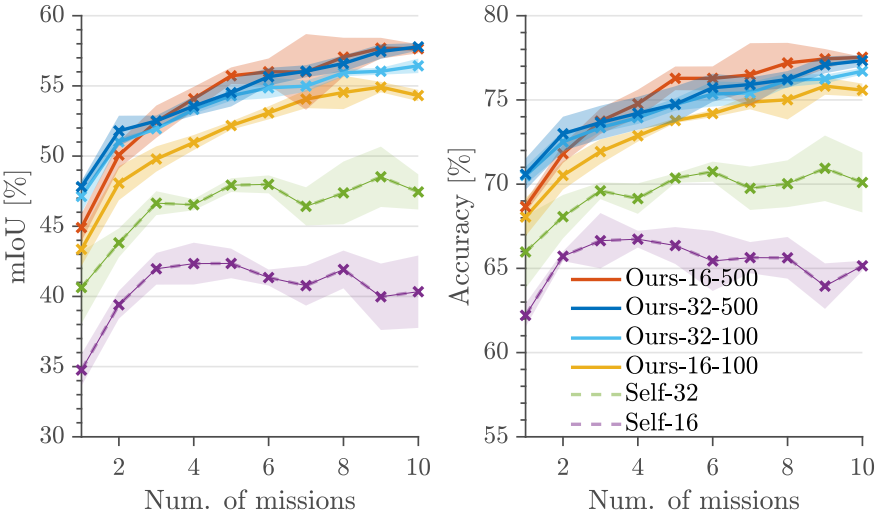


Figure 7.9: Comparison of our semi-supervised approach (solid lines) with 500 or 1000 human-labelled pixels per collected image and a self-supervised approach (dashed lines) using 16 or 32 densely human-labelled images for model pre-training. Results are averaged over three runs on ISPRS Potsdam. Shaded regions indicate one standard deviation. Our semi-supervised approach outperforms the self-supervised approach for all labelling budget configurations.

Although self-supervised robotic active learning approaches do not require any human labelling during deployment, they are inherently limited by their lack of knowledge and systematic prediction errors in unknown terrains, as indicated by our experimental results and discussed by Chaplot et al. [20].

7.4 Conclusion

The increasingly versatile usage of robots for various terrain monitoring tasks, such as precision agriculture and urban monitoring, requires robots to semantically perceive and understand the environment in which they are deployed. Most robotic semantic vision systems use deep learning-based semantic segmentation models trained on static curated human-labelled datasets. As robots are deployed in unknown environments, images recorded during a mission often deviate from the ones the vision system was trained on. This typically leads to a drop in semantic segmentation performance during deployment when using deep learning-based vision systems. To tackle this issue, our adaptive IPP approach for active learning presented in Chap. 6 maximises semantic segmentation performance in unknown environments while reducing the number of human-labelled images. However, it still requires dense pixel-wise human annotation of collected training images, which induces substantial human labelling efforts.

To address this issue, in this chapter, we investigated the thesis' third research question of which image queries should be labelled by a human annotator and which vision learning signals can be derived from the robot's actively im-

proving understanding of the environment to reduce the human labelling effort further. Our main contribution in this chapter is a novel semi-supervised robotic active learning approach that adaptively replans paths to collect new informative training data. A key aspect of our approach is a new semi-supervised semantic vision model training strategy that combines sparse human and pseudo labels. We propose a new method that selects sparse sets of pixels from collected images for human labelling, targeting diverse image regions in which the semantic model predictions are cluttered, thus likely containing informative labelling queries. Furthermore, we introduce a new method that automatically generates pseudo labels rendered from an online-built semantic map in diverse areas of low model uncertainty, likely containing high-quality pseudo labels.

Our experimental results show that our sparse human-labelled pixel selection method outperforms state-of-the-art pixel selection methods in the context of robotic active learning and drastically improves semantic segmentation performance over non-targeted pixel selection for low human labelling budgets. Combining our sparse human labels with our uncertainty-aware pseudo label generation further improves performance. Overall, our semi-supervised robotic active learning approach drastically reduces human labelling requirements compared to the fully supervised framework presented in Chap. 6 while preserving similar semantic segmentation performance. Additionally, our semi-supervised method outperforms purely self-supervised robotic active learning approaches.

These findings demonstrate that, in response to the third research question posed in this thesis, integrating adaptive IPP with semi-supervised learning reduces human labelling efforts and ensures competitive vision model performance in semantic terrain monitoring. Our targeted sparse human label selection method provides one solution towards more efficient image queries. Moreover, our uncertainty-aware method that automatically renders pseudo labels from an online-built semantic map showcases one possible way to use the robot's understanding of the environment to make labelling more efficient. As part of the next chapter, we discuss other possible pathways to answer our third research question and further improve robotic active learning approaches.

Chapter 8

Conclusion

Gathering information autonomously in unknown environments using onboard sensors for data collection is an important skill for robots. Real-world applications include environmental monitoring [38, 172, 179], supporting in search-and-rescue scenarios [36] or planetary exploration missions [117]. Advances in machine learning enable robots to interpret collected sensor measurements automatically [59, 168, 176]. In this way, the robot can perceive its environment and update its understanding by capturing measurements in a map of the environment [80, 114]. To raise the level of autonomy in unknown environments, a key robotic skill is to explore the environment autonomously given resource constraints, such as limited onboard energy and compute power. For this, robots need to decide where to move next despite only having access to noisy and incomplete environmental knowledge.

Our main contributions are novel methods that steer a resource-constrained unmanned aerial vehicle in unknown environments towards areas that yield potentially informative new measurements to improve its understanding of the environment given a limited mission budget, such as flight time. This problem is known as the adaptive informative path planning problem [106, 127]. We have developed novel adaptive informative path planning methods that improve robotic information-gathering efficiency in unknown environments and the robot's semantic understanding while minimising human labelling efforts.

The methods proposed in this thesis contribute to answering our three key research questions: (i) how to increase the compute efficiency of adaptive informative path planning without sacrificing planning performance in information-gathering missions, (ii) how to improve deep learning-based vision models in unknown environments while minimising the amount of human-labelled images for model re-training, and (iii) which image queries should be labelled by a human annotator and which vision learning signals can be derived from the robot's actively improving understanding of the environment to reduce labelling effort?

To approach the first research question, we introduce a learning-based method for adaptive informative path planning that combines classical robotic planning with reinforcement learning to monitor continuous-valued environmental information, such as the surface temperature of arable fields. We perform tree search-based planning during deployment and steer the search using the neural networks offline-learned in simulated monitoring missions. Our learning-based method accelerates the adaptive replanning of paths during deployment compared to compute-expensive non-learning-based adaptive informative path planning methods while showing competitive performance. These results could facilitate deploying adaptive methods on resource-constrained mobile robots.

Most adaptive informative path planning methods are specifically designed for robotic information-gathering missions with different map representations, either monitoring continuous-valued environmental information, such as surface temperatures, or monitoring discrete-valued environmental information, such as semantic segmentation of crops and weeds in arable fields. Hence, changes in the to-be-monitored information require re-designing or even re-training the planning strategy. The second approach addresses this limitation by introducing a novel mathematical formulation of the adaptive informative path planning problem that unifies monitoring missions across various map representations into one map-agnostic planning state representation. Using this state representation and a new reward function for adaptive path planning, the robot learns a single map-agnostic adaptive informative path planning strategy applicable to missions with different to-be-monitored information using reinforcement learning. Our method facilitates deploying learned adaptive path planning strategies without re-designing and re-training them for specific mission characteristics. Additionally, our formulation unifies previously developed planning algorithms while maintaining their performance. In this way, our second approach makes learning-based adaptive informative path planning methods a promising solution to our first research question beyond narrowly defined information-gathering mission characteristics.

Monitoring missions that require a semantic understanding of the environment, e.g. semantically segmenting an urban area into streets, buildings and vegetation for urban planning purposes, typically achieve this using deep learning-based semantic vision models, which process the collected images. These semantic vision models are commonly trained on static human-labelled datasets. As the environment in which the robot is deployed is unknown, images collected during deployment might differ from the ones the model is trained on. In these cases, deep learning-based vision models degrade semantic segmentation performance, requiring costly human annotations of collected images to improve the vision model upon re-training. We propose a novel adaptive informative path planning framework for active learning of robotic vision in semantic monitoring missions

to tackle this issue. Our method's key idea is to link model uncertainty measures from active learning with the information-gathering planning objective. In this way, we adaptively guide the unmanned aerial vehicle to collect potentially informative images for human labelling based on which the model is re-trained. Our framework reduces the number of human-labelled images and, hence, labelling costs for model re-training required to reach competitive semantic segmentation performance compared to prior state-of-the-art and traditional non-adaptive data collection campaigns for semantic vision model training. This verifies that combining adaptive informative path planning methods with active learning of semantic robotic vision is one solution to our second research question.

Lastly, we integrate our adaptive informative path planning for active learning approach with a novel semi-supervised learning framework for label-efficient semantic robotic vision training to further reduce human labelling efforts in semantic monitoring missions. We propose combining sparsely selected informative pixels from collected images for human labelling with automatically labelled pixels using the robot's understanding of the environment. First, we present a new pixel selection method for human labelling that ensures higher semantic segmentation performance with fewer labelled pixels than state-of-the-art methods. Second, we introduce a new method to automatically render semantic labels from areas of the robot's semantic environment map built during deployment in which the model is highly certain about its predictions. Combining our automatically labelled data with human-labelled data for model re-training constantly improves the semantic segmentation performance. Our adaptive informative path planning method for semi-supervised active learning of semantic robotic vision reduces the required human-labelled pixels for model re-training by two magnitudes while ensuring similar performance compared to fully supervised methods in unmanned aerial vehicle-based semantic monitoring missions. Thus, our semi-supervised learning method combined with adaptive informative path planning is one label-efficient solution to semantic robotic vision as an answer to our third research question.

Overall, our overarching research hypothesis was that integrated robot learning and planning approaches can enhance information-gathering efficiency in unmanned aerial vehicle-based monitoring missions. Although our novel methods present only a few of potentially many pathways to integrate or combine learning-based approaches with robotic planning methods, we make crucial contributions to integrated learning and planning for robotic information gathering in various applications. Our first two approaches show how to combine classical robotic path planning and reinforcement learning to increase information-gathering efficiency by advancing and unifying learning-based adaptive informative path planning methods. Furthermore, our third and fourth methods connect the field of active learning in computer vision and adaptive informative path planning algorithms

with semi-supervised learning techniques in a novel fashion. These established connections between both research fields yield novel methods that increase the robot's semantic vision performance while minimising human labelling effort. Overall, the methods and experimental results presented in this thesis suggest that integrating learning-based approaches and planning methods more closely indeed enhances robotic information-gathering efficiency in various applications.

8.1 Future Work

In this thesis, we investigated approaches for robotic information gathering in unknown environments. We presented multiple new methods for adaptive IPP enabling efficient collection of information using resource-constrained robots. Despite the encouraging performance of our introduced methods and their improvements over previous state-of-the-art approaches, future research to further improve the efficiency of robotic information gathering is required to achieve higher levels of robot autonomy. In Sec. 8.1.1, we discuss open research questions and possible future directions to improve the training and generalisability of learning-based adaptive IPP methods, such as the ones proposed in Chap. 4 and Chap. 5. In Sec. 8.1.2, we discuss open research questions and possible future directions to extend our approaches to active learning of robotic vision introduced in Chap. 6 and Chap. 7 and reduce their human labelling requirements further.

This section incorporates material from the following peer-reviewed conference workshop publication, for which I have been the main contributor:

- Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Active Learning of Robot Vision Using Adaptive Path Planning. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS) Workshop on Label Efficient Learning Paradigms for Autonomy at Scale*, 2024

8.1.1 Learning-Based Adaptive Informative Path Planning

In the following, we discuss open challenges in our RL-based adaptive IPP approaches introduced in Chap. 4 and Chap. 5. We identify open research questions in planning policy training and their applications to unstructured environments and multi-robot teams. We suggest future directions for addressing them.

Sample-Efficient Training of Planning Policies. Despite their promising performance, our adaptive IPP policies learned using RL require substantial amounts of simulated missions during training. For our method discussed in Chap. 5, this results in more than two days of policy training on a single GPU

workstation. We train policies offline and perform policy inference during deployment. Thus, high training times do not necessarily affect deployment frequencies. However, the sample complexity of training planning policies with RL is expected to grow significantly with increasingly complex missions in more complex environments, e.g. 3D environments with obstacles causing occlusions.

A promising direction to train planning policies in a more sample-efficient fashion is to use imitation learning methods. Choudhury et al. [29] introduce an imitation learning method to train exploration policies that imitate an expert planner. The expert planner has access to the privileged ground truth information of the environment, while the learned policy is trained to imitate this expert planner based on partial information available during deployment. Similarly, future work could investigate using imitation learning methods, such as behaviour cloning [123], to train policies for our more challenging adaptive IPP problem. One could readily use our map-agnostic adaptive IPP formulation and non-learning-based expert planners, such as the map-agnostic Monte Carlo tree search planning algorithm introduced in Chap. 5, to generate a dataset of expert paths and their associated information values. Based on this dataset, an actor-critic network can be trained using imitation learning. The actor-critic network could also be integrated as a learned policy and value function steering the sampling-based adaptive IPP algorithm proposed in Chap. 4. Last, one could use actor-critic RL algorithms to fine-tune the pre-trained policy [98] to further improve the performance of policies trained with imitation learning.

Unstructured 3D Environments. This thesis focuses mainly on information gathering with UAVs monitoring large but flat 2D terrain information. In general, our adaptive IPP approaches discussed in Chap. 4 and Chap. 5 also apply to other robot platforms, onboard sensors and obstacle-free 3D environments, e.g. to unmanned ground vehicle-based indoor temperature mapping scenarios [125]. However, our learning-based adaptive IPP methods must be adapted to extend the range of robotic information-gathering missions to unstructured 3D environments with obstacles that result in occlusions. In our joint work led by Apoorva Vashisth [167], we introduce a learned adaptive IPP policy for deployment on a UAV in unstructured 3D environments trained using RL. We introduce a topological graph representing the action space around the UAV. The graph captures the collision-free workspace of the UAV to ensure obstacle avoidance in static environments. Furthermore, we propose a new reward function for viewpoint planning to maximise newly collected information. Our experimental results show that both changes to previous methods are necessary to reliably avoid collisions and to maximise information-gathering performance in 3D environments.

However, these adaptive IPP methods assume static environments. These approaches either expect obstacle-free robot workspaces or static obstacles that

do not move during mission time. Moreover, all approaches assume that the to-be-mapped information of interest, e.g. surface temperature, varies only spatially across the environment but does not change over time for a fixed point in the environment. Both limitations are important to address in future methods. The assumption of temporally static information is often only valid for monitoring missions with short mission times compared to the rate of change of the information. For example, monitoring fruits in an orchard is assumed to be temporally static as fruit growth is slow. In contrast, monitored temperatures of metallic surfaces might change rapidly depending on the environmental conditions. Additionally, enabling the robot to reason about and avoid collisions with dynamically moving objects could enable missions that track target objects in an environment. Dynamic object collision avoidance particularly enables deploying adaptive IPP methods in environments while working alongside other agents, such as human workers, remote-controlled machines or other autonomous robots.

Heterogeneous Multi-Robot Teams. We focused on robotic information gathering with a single robot. In large-scale monitoring missions, such as monitoring larger regions of oceans or forests, deploying a team of robots instead of a single robot is often advantageous [6, 15]. Robot teams offer improved information-gathering efficiency in larger environments due to better spatial coverage. At the same time, deploying multiple robots reduces the impact of robot failures on the information-gathering performance. Leveraging a team of robots for efficient monitoring comes with new challenges for adaptive IPP methods. A key skill for robot teams is to plan paths cooperatively to maximise the gathered information given a limited mission budget. As we showed in Chap. 4 and Chap. 5, non-learning-based methods for single-robot applications are compute-expensive.

Planning paths cooperatively for robot teams increases replanning times even further as the number of possible paths and, thus, computational complexity scale exponentially with the number of robots. To avoid this issue, we introduce a novel multi-UAV adaptive IPP method that trains planning policies for terrain monitoring in simulation using multi-agent RL in a joint work led by Jonas Westheider [175]. UAVs compute-efficiently plan paths during deployment based on their individual understanding of the environment via offline-learned policy inference while being robust to communication failures with other robots and varying team sizes. However, this and other multi-robot adaptive IPP methods commonly consider homogeneous robot teams in which each robot has the same platform, is equipped with the same sensors and yields the same budget constraints. This limits the efficiency of information gathering since allocating budgets and paths cannot complement each robot's weaknesses. Future methods that plan paths and allocate resources to heterogeneous robot teams might improve the efficiency of large-scale information-gathering missions.

Generalisability of Learned Policies. Learning-based adaptive IPP methods, as those discussed in Chap. 4 and Chap. 5, show great potential due to their compute-efficient planning policy inference during deployment. Furthermore, they maintain competitive information-gathering performance with computationally expensive non-learning-based adaptive IPP methods. However, as with all data-driven methods, our learned planning strategies degrade in performance in cases where the environment strongly deviates from the ones seen during policy training in simulation. As environments are unknown before deployment, we cannot always ensure that environments simulated during training resemble deployment conditions. Thus, unlike non-learning-based planning, learning-based adaptive IPP methods cannot always ensure reliable performance.

An open research question is how to ensure reliable performance in environments that strongly differ from the ones seen during training. To date, this poses a general research question in the RL community. Zero-shot policy inference in unseen environments without adapting the policy to the environment during deployment often performs poorly, not only in adaptive IPP [78]. To advance the generalisation abilities of learning-based adaptive IPP methods, one valuable research question is how to adapt offline-learned planning strategies efficiently during deployment. This research question is closely related to the field of meta-RL. Meta-RL aims to offline-learn an initial policy that can be adapted to new environments and tasks with only a few environment interactions during deployment [54]. This small amount of newly gathered experience must be informative to update the initial policy efficiently. Thus, a promising avenue to develop novel learning-based robotic information-gathering methods could be combining recent advances from the field of meta-RL with adaptive IPP methods.

8.1.2 Active Learning of Robotic Vision Using Adaptive Planning

In the following, we discuss open challenges in our approaches to active learning of robotic vision introduced in Chap. 6 and Chap. 7. We identify open research questions in self-supervision, uncertainty quantification, continual learning and model training efficiency and suggest future directions to address them.

Larger Variety of Applications & Tasks. Current adaptive robotic planning methods for active learning consider applications and vision tasks with limited variety in their experimental evaluation. Methods that rely on self-supervised learning to improve semantic vision evaluate performance in 3D indoor household scenarios using unmanned ground vehicles. Environments used for pre-training and environments encountered during deployment are often similar in visual appearance [20, 45, 191]. In contrast, adaptive planning methods that rely on

human annotations are exposed to larger visual variations. However, they are only evaluated in UAV-based semantic 2D terrain mapping [12, 134, 136, 138]. For a more challenging and standardised evaluation of methods, it would be beneficial to deploy and evaluate methods in vastly varying 3D outdoor and indoor environments using varying robot platforms. Furthermore, most methods aim to improve semantic segmentation performance in unknown environments [12, 45, 136, 138, 191]. To enable a broader spectrum of downstream manipulation and intervention tasks, the robotic vision should also segment and associate individual object instances with their semantics and semantically segment other scene parts, i.e. performing panoptic segmentation. This might require integrating new mapping methods supporting fusing panoptic predictions [143] to create self-supervision signals. Learning panoptic segmentation also requires new, efficient human labelling methods beyond costly pixel-wise labels.

Novel Embodied Self-Supervised Learning Methods. Fully and semi-supervised active learning methods are still impacted by costly human labelling requirements. Next to accelerating the human labelling process itself, high-quality self-supervised learning is required to keep the amount of human labelling queries low and, at the same time, reach maximal prediction performance. Self-supervised [20, 45, 191] and our semi-supervised method in Chap. 7 create pseudo labels from an online-built semantic map. These methods render pseudo labels from voxel-based maps containing fused semantic predictions from viewpoints encountered during deployment. However, voxel-based maps cannot render image-label pairs from novel viewpoints. Rendering additionally suffers from discretisation artefacts. Recently, semantic neural rendering approaches enhanced self-supervised pseudo label quality and diversity. These methods render high-quality image-label pairs from novel viewpoints not encountered during deployment. Neural rendering methods have been shown to outperform voxel map-based self-supervision [97]. Jin et al. [69] combine semantic neural rendering with adaptive planning of informative next viewpoints for semantic-geometric single object reconstruction. Similarly, combining self-supervised semantic neural rendering with adaptive planning for active learning of robotic vision could improve the current systems' vision performances without additional human labels.

Improved Uncertainty Quantification. As discussed by Chaplot et al. [20], overconfidently wrong predictions could lead to reinforced prediction errors after re-training on these predictions in a self-supervised fashion. Even our human-guided methods require well-calibrated model uncertainty estimation to maximise the prediction performance while minimising human labelling effort. Thus, better-calibrated model uncertainty estimation techniques are required as current state-of-the-art methods still tend to produce overconfident predictions [8, 46, 128]. Moreover, adaptive planning methods ignore various sources of un-

certainty. All methods use some form of model uncertainty [191] or prediction confidence [20] to collect potentially informative training data. Future research could integrate and disentangle other sources of uncertainty, such as data uncertainty [76] induced by environmental factors or noisy sensors. This information could be used to avoid requesting human labels for images with high data uncertainty that contribute little to the model improvements [76]. Additionally, adaptively planning viewpoints towards hard-to-predict objects due to environmental factors might reduce sources of data uncertainty [115].

Towards Continual Active Learning. Another key challenge for efficient active learning of robotic vision systems is the robot's ability to continually learn about new unseen environments. Continual learning requires transferring the knowledge gained during previous deployments in different environments [91] without suffering from catastrophic forgetting due to changing training data distributions [103]. This problem of continual learning is largely ignored in our and other adaptive planning methods for active learning of robotic vision. To the best of our knowledge, Frey et al. [45] proposed the only continual learning method for active learning of robotic vision to date. However, they do not leverage adaptive planning for training data collection; instead, they treat the robot as a passive data collection device. Furthermore, their approach is constrained to indoor environments with a fixed set of known semantic classes, rendering it impractical for semantics varying across environments. Combining adaptive planning with continual learning in various environments could lead to more robust vision systems. Additionally, it could result in a more targeted continuous collection of informative training data while leveraging already gained previous knowledge. In this way, the robotic vision potentially generalises better to unseen environments over time while requiring progressively fewer human labels.

Improved Model Training Efficiency. Our and other adaptive replanning methods for active learning [12, 20, 191] iteratively re-train the vision model after a mission is finished, required to efficiently adapt the robot's data collection based on previously collected training data influencing the model performance. Although our proposed methods use relatively lightweight network architectures, iterative re-training is prohibitively expensive in applications that require fast online adaption of vision or frequent re-deployment cycles. One way to improve the network re-training efficiency could be to leverage recent progress in vision foundation models [77]. These foundation models could serve as pre-trained frozen semantic feature extractors. Additionally, they could be combined with small, lightweight, trainable adapter networks for active learning of robotic vision [66]. This could mitigate the costly re-training of larger networks while allowing the robotic vision to profit from zero- and few-shot generalisation advances in vision foundation models likely to transfer to our robotic active learning setting.

Bibliography

[1] Alireza Ahmadi, Michael Halstead, and Chris McCool. BonnBot-I: A Precise Weed Management and Crop Monitoring Platform. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.

[2] Alireza Ahmadi, Julius Rückin, Michael Halstead, Marija Popović, and Chris McCool. OptimWeeder: A Reinforcement Learning-Based Approach to Control Mobile Multi-Axes Weeding Systems. *Computers and Electronics in Agriculture, under review*, 2024.

[3] Akash Arora, P. Michael Furlong, Robert Fitch, Salah Sukkarieh, and Terrence Fong. Multi-modal Active Perception for Information Gathering in Science Missions. *Autonomous Robots*, 43:1827–1853, 2019.

[4] Shi Bai, Fanfei Chen, and Brendan Englot. Toward Autonomous Mapping and Exploration for Mobile Robots through Deep Supervised Learning. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[5] Ruzena Bajcsy. Active perception. *Proc. of the IEEE*, 76(8):966–1005, 1988.

[6] Joseph L. Baxter, Edmund K. Burke, Jonathan M. Garibaldi, and Mark Norman. *Multi-robot Search and Rescue: A Potential Field based Approach.* Springer, 2007.

[7] Avital Bechar and Clément Vigneault. Agricultural robots for field operations: Concepts and components. *Biosystems Engineering*, 149:94–111, 2016.

[8] William H. Beluch, Tim Genewein, Andreas Nürnberger, and Jan M. Köhler. The Power of Ensembles for Active Learning in Image Classification. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[9] Rodrigo Benenson and Vittorio Ferrari. From Colouring-in to Pointillism: Revisiting Semantic Segmentation Supervision. *arXiv preprint, arXiv:2210.14142*, 2022.

[10] Dimitri Bertsekas. *Dynamic Programming and Optimal Control*, volume 4. Athena Scientific, 2012.

[11] Graeme Best, Oliver M. Cliff, Timothy Patten, Ramgopal R. Mettu, and Robert Fitch. Dec-MCTS: Decentralized Planning for Multi-robot Active

Perception. *Intl. Journal of Robotics Research (IJRR)*, 38(2-3):316–337, 2019.

[12] Hermann Blum, Silvan Rohrbach, Marija Popović, Luca Bartolomei, and Roland Siegwart. Active Learning for UAV-based Semantic Mapping. In *Proc. of Robotics: Science and Systems Workshop on Informative Path Planning and Adaptive Sampling*, 2019.

[13] Cameron B. Browne, Edward Powley, Daniel Whitehouse, Simon M. Lucas, Peter I. Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A Survey of Monte Carlo Tree Search Methods. *IEEE Trans. on Computational Intelligence and AI in Games*, 4(1):1–43, 2012.

[14] Bernadette Bucher, Karl Schmeckpeper, Nikolai Matni, and Kostas Daniilidis. An Adversarial Objective for Scalable Exploration. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021.

[15] Wolfram Burgard, Mark Moors, Cyrill Stachniss, and Frank E. Schneider. Coordinated Multi-Robot Exploration. *IEEE Trans. on Robotics (TRO)*, 21(3):376–378, 2005.

[16] Wolfram Burgard, Cyrill Stachniss, and Giorgio Grisetti. Information Gain-based Exploration Using Rao-Blackwellized Particle Filters. In *Proc. of the Learning Workshop (Snowbird)*, 2005.

[17] Colin Campbell, Nello Cristianini, and Alex Smola. Query Learning with Large Margin Classifiers. In *Proc. of the Intl. Conf. on Machine Learning (ICML)*, 2000.

[18] Yuhong Cao, Tianxiang Hou, Yizhuo Wang, Xian Yi, and Guillaume Sartoretti. ARiADNE: A Reinforcement Learning Approach using Attention-based Deep Networks for Exploration. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[19] Yuhong Cao, Yizhuo Wang, Apoorva Vashisth, Haolin Fan, and Guillaume Adrien Sartoretti. CAtNIPP: Context-Aware Attention-based Network for Informative Path Planning. In *Proc. of the Conf. on Robot Learning (CoRL)*, 2023.

[20] Devendra S. Chaplot, Murtaza Dalal, Saurabh Gupta, Jitendra Malik, and Russ R. Salakhutdinov. SEAL: Self-supervised Embodied Active Learning using Exploration and 3D Consistency. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2021.

[21] Arthur Charpentier, Romuald Elie, and Carl Remlinger. Reinforcement Learning in Economics and Finance. *Computational Economics*, 62:425—462, 2023.

[22] Fanfei Chen, Shi Bai, Tixiao Shan, and Brendan Englot. Self-Learning Exploration and Mapping for Mobile Robots via Deep Reinforcement Learning. In *Proc. of the AIAA SciTech Forum*, 2019.

[23] Fanfei Chen, John D. Martin, Yewei Huang, Jinkun Wang, and Brendan Englot. Autonomous Exploration Under Uncertainty via Deep Reinforcement Learning on Graphs. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[24] Fanfei Chen, Paul Szenher, Yewei Huang, Jinkun Wang, Tixiao Shan, Shi Bai, and Brendan Englot. Zero-Shot Reinforcement Learning on Graphs for Autonomous Exploration Under Uncertainty. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2021.

[25] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big Self-Supervised Models are Strong Semi-Supervised Learners. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2020.

[26] Nuttapong Chentanez, Andrew Barto, and Satinder Singh. Intrinsically motivated reinforcement learning. In *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, 2004.

[27] Taeyeong Choi and Grzegorz Cielniak. Adaptive Selection of Informative Path Planning Strategies via Reinforcement Learning. In *Proc. of the Europ. Conf. on Mobile Robotics (ECMR)*, 2021.

[28] Howie Choset. Coverage for Robotics - A Survey of Recent Results. *Annals of mathematics and artificial intelligence*, 31(1):113–126, 2001.

[29] Sanjiban Choudhury, Mohak Bhardwaj, Sankalp Arora, Ashish Kapoor, Gireeja Ranade, Sebastian Scherer, and Debadeepta Dey. Data-driven Planning via Imitation Learning. *Intl. Journal of Robotics Research (IJRR)*, 37(13-14):1632–1672, 2018.

[30] Shushman Choudhury, Nate Gruver, and Mykel J. Kochenderfer. Adaptive Informative Path Planning with Multimodal Sensing. In *Proc. of the Int. Conf. on Automated Planning and Scheduling (ICAPS)*, 2020.

[31] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele.

The Cityscapes Dataset for Semantic Urban Scene Understanding. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[32] Xu-Yang Dai, Qing-Hao Meng, Sheng Jin, and Yin-Bo Liu. Camera view planning based on generative adversarial imitation learning in indoor active exploration. *Applied Soft Computing*, 129:109621, 2022.

[33] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[34] Christopher E. Denniston, Gautam Salhotra, Akseli Kangaslahti, David A. Caron, and Gaurav S. Sukhatme. Learned Parameter Selection for Robotic Information Gathering. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[35] Harnaik Dhami, Vishnu D. Sharma, and Pratap Tokekar. Pred-NBV: Prediction-guided Next-Best-View Planning for 3D Object Reconstruction. *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[36] Patrick Doherty and Piotr Rudol. A UAV Search and Rescue Scenario with Human Body Detection and Geolocalization. In *Australasian Joint Conf. on Artificial Intelligence*, 2007.

[37] Louis Dressel and Mykel J. Kochenderfer. On the Optimality of Ergodic Trajectories for Information Gathering Tasks. In *Annual American Control Conference (ACC)*, 2018.

[38] Matthew Dunbabin and Lino Marques. Robots for Environmental Monitoring: Significant Advancements and Applications. *IEEE Robotics and Automation Magazine (RAM)*, 19(1):24–39, 2012.

[39] Nikita Durasov, Timur Bagautdinov, Pierre Baque, and Pascal Fua. Masksembles for Uncertainty Estimation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[40] Sayna Ebrahimi, William Gan, Dian Chen, Giscard Biamby, Kamyar Salahi, Michael Laielli, Shizhan Zhu, and Trevor Darrell. Minimax Active Learning. *arXiv preprint, arXiv:2012.10467*, 2020.

[41] Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, Shane

Legg, and Koray Kavukcuoglu. IMPALA: Scalable Distributed Deep-RL with Importance Weighted Actor-Learner Architectures. In *Proc. of the Intl. Conf. on Machine Learning (ICML)*, 2018.

[42] Mark Everingham, Luc Van Gool, Christopher K.I. Williams, John Winn, and Andrew Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *Intl. Journal of Computer Vision (IJCV)*, 88(2):303–338, 2010.

[43] Michael Firman. RGBD Datasets: Past, Present and Future. In *CVPR Workshop on Large Scale 3D Data: Acquisition, Modelling and Analysis*, 2016.

[44] Yoav Freund, H Sebastian Seung, Eli Shamir, and Naftali Tishby. Selective Sampling Using the Query by Committee Algorithm. *Machine Learning*, 28(2):133–168, 1997.

[45] Jonas Frey, Hermann Blum, Francesco Milano, Roland Siegwart, and Cesar Cadena. Continual Adaptation of Semantic Segmentation using Complementary 2D-3D Data Representations. *IEEE Robotics and Automation Letters (RA-L)*, 7(4):11665–11672, 2022.

[46] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *Proc. of the Intl. Conf. on Machine Learning (ICML)*, 2016.

[47] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. Deep Bayesian Active Learning with Image Data. In *Proc. of the Intl. Conf. on Machine Learning (ICML)*, 2017.

[48] Enric Galceran and Marc Carreras. A Survey on Coverage Path Planning for Robotics. *Robotics and Autonomous Systems*, 61(12):1258–1276, 2013.

[49] Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, and José García Rodríguez. A Review on Deep Learning Techniques Applied to Semantic Segmentation. *arXiv preprint, arXiv:1704.06857*, 2017.

[50] Georgios Georgakis, Bernadette Bucher, Anton Arapin, Karl Schmeckpeper, Nikolai Matni, and Kostas Daniilidis. Uncertainty-driven Planner for Exploration and Navigation. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[51] Georgios Georgakis, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, and Kostas Daniilidis. Learning to Map for Active Semantic Goal

Navigation. In *Proc. of the Int. Conf. on Learning Representations (ICLR)*, 2022.

[52] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On Calibration of Modern Neural Networks. In *Proc. of the Intl. Conf. on Machine Learning (ICML)*, 2017.

[53] Yuhong Guo and Dale Schuurmans. Discriminative Batch Mode Active Learning. In *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, 2007.

[54] Abhishek Gupta, Russell Mendonca, YuXuan Liu, Pieter Abbeel, and Sergey Levine. Meta-Reinforcement Learning of Structured Exploration Strategies. In *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, 2018.

[55] Florian Görlich, Elias Marks, Anne-Katrin Mahlein, Kathrin König, Philipp Lottes, and Cyrill Stachniss. UAV-Based Classification of Cercospora Leaf Spot Using RGB Images. *Drones*, 5(2):34, 2021.

[56] David Hall, Feras Dayoub, Jason Kulk, and Chris McCool. Towards Unsupervised Weed Scouting for Agricultural Robotics. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2017.

[57] Nikolaus Hansen. The CMA Evolution Strategy: A Comparing Review. *Towards a new evolutionary computation: Advances in the estimation of distribution algorithms*, pages 75–102, 2006.

[58] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Trans. on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[59] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[60] Ruifei He, Jihan Yang, and Xiaojuan Qi. Re-distributing Biased Pseudo Labels for Semi-supervised Semantic Segmentation: A Baseline Investigation. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[61] Gregory Hitz, Enric Galceran, Marie-Ève Garneau, François Pomerleau, and Roland Siegwart. Adaptive Continuous-Space Informative Path Planning for Online Environmental Monitoring. *Journal of Field Robotics (JFR)*, 34(8):1427–1449, 2017.

[62] David Hoeller, Nikita Rudin, Dhionis Sako, and Marco Hutter. Anymal Parkour: Learning Agile Navigation for Quadrupedal Robots. *Science Robotics*, 9(88):eadi7566, 2024.

[63] Geoffrey A. Hollinger and Gaurav S. Sukhatme. Sampling-based Robotic Information Gathering Algorithms. *Intl. Journal of Robotics Research (IJRR)*, 33(9):1271–1287, 2014.

[64] Geoffrey A. Hollinger, Brendan Englot, Franz S. Hover, Urbashi Mitra, and Gaurav S. Sukhatme. Active Planning for Underwater Inspection and the Benefit of Adaptivity. *Intl. Journal of Robotics Research (IJRR)*, 32(1): 3–18, 2013.

[65] Neil Houlsby, Ferenc Huszár, Zoubin Ghahramani, and Máté Lengyel. Bayesian Active Learning for Classification and Preference Learning. *arXiv preprint, arXiv:1112.5745*, 2011.

[66] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. In *Proc. of the Int. Conf. on Learning Representations (ICLR)*, 2022.

[67] Gao Huang, Yixuan Li, Geoff Pleiss, Zhuang Liu, John E. Hopcroft, and Kilian Q. Weinberger. Snapshot Ensembles: Train 1, Get M for Free. In *Proc. of the Int. Conf. on Learning Representations (ICLR)*, 2017.

[68] ISPRS. 2D Semantic Labeling Contest, 2018. URL `https://www.isprs.org/education/benchmarks/UrbanSemLab/semantic-labeling.aspx`.

[69] Liren Jin, Haofei Kuang, Yue Pan, Cyrill Stachniss, and Marija Popović. STAIR: Semantic-Targeted Active Implicit Reconstruction. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2024.

[70] Ajay J. Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-class Active Learning for Image Classification. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[71] Ashish Kapoor, Kristen Grauman, Raquel Urtasun, and Trevor Darrell. Active Learning with Gaussian Processes for Object Categorization. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2007.

[72] Elia Kaufmann, Leonard Bauersfeld, Antonio Loquercio, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. Champion-level Drone Racing using Deep Reinforcement Learning. *Nature*, 620(7976):982–987, 2023.

[73] Lydia E. Kavraki, Petr Svestka, J.-C. Latombe, and Mark H. Overmars. Probabilistic Roadmaps for Path Planning in High-dimensional Configuration Spaces. *IEEE Trans. on Robotics and Automation*, 12(4):566–580, 1996.

[74] Benjamin Kellenberger, Diego Marcos, Sylvain Lobry, and Devis Tuia. Half a Percent of Labels is Enough: Efficient Animal Detection in UAV Imagery Using Deep CNNs and Active Learning. *IEEE Trans. on Geoscience and Remote Sensing*, 57(12):9524–9533, 2019.

[75] Ronald Kemker, Carl Salvaggio, and Christopher Kanan. Algorithms for Semantic Segmentation of Multispectral Remote Sensing Imagery Using Deep Learning. *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)*, 145:60–77, 2018.

[76] Alex Kendall and Yarin Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, 2017.

[77] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment Anything. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2023.

[78] Robert Kirk, Amy Zhang, Edward Grefenstette, and Tim Rocktäschel. A Survey of Zero-shot Generalisation in Deep Reinforcement Learning. *Journal of Artificial Intelligence Research (JAIR)*, 76, 2023.

[79] Mykel J. Kochenderfer, Tim A. Wheeler, and Kyle H. Wray. *Algorithms for decision making*. MIT press, 2022.

[80] Ioannis Kostavelis and Antonios Gasteratos. Semantic mapping for mobile robotics tasks: A survey. *Journal on Robotics and Autonomous Systems (RAS)*, 66:86–103, 2015.

[81] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, 2012.

[82] James J. Kuffner and Steven M. LaValle. RRT-connect: An efficient approach to single-query path planning. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2000.

[83] K. Niranjan Kumar, Irfan Essa, and Sehoon Ha. Graph-based Cluttered Scene Generation and Interactive Exploration using Deep Reinforcement

Learning. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[84] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and Scalable Predictive Uncertainty Estimation Using Deep Ensembles. *Proc. of the Advances in Neural Information Processing Systems (NIPS)*, 2017.

[85] Ferdinand Langer, Andres Milioto, Alexandre Haag, Jens Behley, and Cyrill Stachniss. Domain Transfer for Semantic Segmentation of LiDAR Data using Deep Neural Networks. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[86] David Lattanzi and Gregory Miller. Review of Robotic Infrastructure Inspection Systems. *Journal of Infrastructure Systems*, 23(3):04017004, 2017.

[87] Steven LaValle. Rapidly-exploring random trees: A new tool for path planning. Technical report, Iowa State University, 1998.

[88] Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning Quadrupedal Locomotion over Challenging Terrain. *Science robotics*, 5(47):eabc5986, 2020.

[89] Camille C.D. Lelong, Philippe Burger, Guillaume Jubelin, Bruno Roux, Sylvain Labbé, and Frédéric Baret. Assessment of Unmanned Aerial Vehicles Imagery for Quantitative Monitoring of Wheat Crop in Small Plots. *Sensors*, 8(5):3557–3585, 2008.

[90] Gaston Lenczner, Adrien Chan-Hon-Tong, Bertrand Le Saux, Nicola Luminari, and Guy Le Besnerais. DIAL: Deep Interactive and Active Learning for Semantic Segmentation in Remote Sensing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:3376–3389, 2022.

[91] Timothée Lesort, Vincenzo Lomonaco, Andrei Stoian, Davide Maltoni, David Filliat, and Natalia Díaz-Rodríguez. Continual Learning for Robotics: Definition, Framework, Learning Strategies, Opportunities and Challenges. *Information Fusion*, 58:52–68, 2020.

[92] David D. Lewis and William A. Gale. A Sequential Algorithm for Training Text Classifiers. In *Proc. of the Int. ACM-SIGIR Conf. on Research and Development in Information Retrieval*, 1994.

[93] Xin Li and Yuhong Guo. Adaptive Active Learning for Image Classification. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[94] Yimeng Li, Arnab Debnath, Gregory J. Stein, and Jana Košecká. Learning-Augmented Model-Based Planning for Visual Exploration. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[95] Zhengqi Li and Volkan Isler. Large Scale Image Mosaic Construction for Agricultural Applications. *IEEE Robotics and Automation Letters (RA-L)*, 1(1):295–302, 2016.

[96] Ruishan Liu and James Zou. The Effects of Memory Replay in Reinforcement Learning. In *Proc. of Allerton Conf. on Communication, Control, and Computing*, 2018.

[97] Zhizheng Liu, Francesco Milano, Jonas Frey, Roland Siegwart, Hermann Blum, and Cesar Cadena. Unsupervised Continual Semantic Adaptation through Neural Rendering. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[98] Zuxin Liu, Mohit Deshpande, Tony Qi, Ding Zhao, Rajasimman Madhivanan, and Arnie Sen. Learning to Explore (L2E): Deep Reinforcement Learning-based Autonomous Exploration for Household Robot. In *Proc. of the RSS Workshop on Robot Representations for Scene Understanding, Reasoning, and Planning*, 2023.

[99] Iker Lluvia, Elena Lazkano, and Ander Ansuategi. Active Mapping and Robot Exploration: A Survey. *Sensors*, 21(7), 2021.

[100] Max Lodel, Bruno Brito, Alvaro Serra-Gómez, Laura Ferranti, Robert Babuška, and Javier Alonso-Mora. Where to Look Next: Learning Viewpoint Recommendations for Informative Trajectory Planning. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[101] Amit Mandelbaum and Daphna Weinshall. Distance-based Confidence Score for Neural Network Classifiers. *arXiv preprint, arXiv:1709.09844*, 2017.

[102] Roman Marchant, Fabio Ramos, and Scott Sanner. Sequential Bayesian Optimisation for Spatial-Temporal Monitoring. In *Proc. of the Conf. on Uncertainty in Artificial Intelligence (UAI)*, 2014.

[103] Michael McCloskey and Neal J. Cohen. Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. In *Psychology of Learning and Motivation*, volume 24, pages 109–165. Academic Press, 1989.

[104] Chris McCool, James Beattie, Jennifer Firn, Chris Lehnert, Jason Kulk, Owen Bawden, Raymond Russell, and Tristan Perez. Efficacy of Mechanical Weeding Tools: A Study into Alternative Weed Management Strategies Enabled by Robotics. *IEEE Robotics and Automation Letters (RA-L)*, 3 (2):1184–1190, 2018.

[105] Ajith Anil Meera, Marija Popović, Alexander Millane, and Roland Siegwart. Obstacle-aware Adaptive Informative Path Planning for UAV-based Target Search. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2019.

[106] Alexandra Meliou, Andreas Krause, Carlos Guestrin, and Joseph M. Hellerstein. Nonmyopic Informative Path Planning in Spatio-Temporal Models. In *Proc. of the Conf. on Advancements of Artificial Intelligence (AAAI)*, 2007.

[107] Rohit Menon, Tobias Zaenker, Nils Dengler, and Maren Bennewitz. NBV-SC: Next Best View Planning based on Shape Completion for Fruit Mapping and Reconstruction. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[108] Gonzalo Mier, João Valente, and Sytze de Bruin. Fields2Cover: An Open-Source Coverage Path Planning Library for Unmanned Agricultural Vehicles. *IEEE Robotics and Automation Letters (RA-L)*, 8(4):2166–2172, 2023.

[109] Andres Milioto, Philipp Lottes, and Cyrill Stachniss. Real-time Semantic Segmentation of Crop and Weed for Precision Agriculture Robots Leveraging Background Knowledge in CNNs. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2018.

[110] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. RangeNet++: Fast and Accurate LiDAR Semantic Segmentation. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.

[111] Lauren M. Miller and Todd D. Murphey. Trajectory Optimization for Continuous Ergodic Exploration. In *American Control Conference (ACC)*, 2013.

[112] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level Control through Deep Reinforcement Learning. *Nature*, 518(7540):529–533, 2015.

[113] Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. Monte Carlo Gradient Estimation in Machine Learning. *Journal on Machine Learning Research (JMLR)*, 21(132):1–62, 2020.

[114] Hans Moravec and Alberto Elfes. High resolution maps from wide angle sonar. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 1985.

[115] David Morilla-Cabello, Jonas Westheider, Marija Popović, and Eduardo Montijano. Perceptual Factors for Environmental Modeling in Robotic Active Perception. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2024.

[116] Farzad Niroui, Kaicheng Zhang, Zendai Kashino, and Goldie Nejat. Deep Reinforcement Learning Robot for Search and Rescue Applications: Exploration in Unknown Cluttered Environments. *IEEE Robotics and Automation Letters (RA-L)*, 4(2):610–617, 2019.

[117] Joshua Ott, Edward Balaban, and Mykel J. Kochenderfer. Sequential Bayesian Optimization for Adaptive Informative Path Planning with Multimodal Sensing. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[118] Joshua Ott, Edward Balaban, and Mykel J. Kochenderfer. Trajectory Optimization for Adaptive Informative Path Planning with Multimodal Sensing. In *Proc. of Intl. Conf. on Control, Decision and Information Technologies (CoDIT)*, 2024.

[119] Joshua Ott, Mykel J. Kochenderfer, and Stephen Boyd. Approximate Sequential Optimization for Informative Path Planning. *Journal on Robotics and Autonomous Systems (RAS)*, 182:104814, 2024.

[120] Yassine Ouali, Céline Hudelot, and Myriam Tami. Semi-supervised Semantic Segmentation with Cross-Consistency Training. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[121] Nicolas Papernot and Patrick McDaniel. Deep k-Nearest Neighbors: Towards Confident, Interpretable and Robust Deep Learning. *arXiv preprint, arXiv:1803.04765*, 2018.

[122] Nick Pawlowski, Miguel Jaques, and Ben Glocker. Efficient Variational Bayesian Neural Network Ensembles for Outlier Detection. *arXiv preprint, arXiv:1703.06749*, 2017.

[123] Dean A. Pomerleau. Efficient Training of Artificial Neural Networks for Autonomous Navigation. *Neural Computation*, 3(1):88–97, 1991.

[124] Marija Popović, Gregory Hitz, Juan Nieto, Inkyu Sa, Roland Siegwart, and Enric Galceran. Online Informative Path Planning for Active Classification Using UAVs. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2017.

[125] Marija Popović, Teresa Vidal-Calleja, Jen Jen Chung, Juan Nieto, and Roland Siegwart. Informative Path Planning for Active Field Mapping under Localization Uncertainty. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[126] Marija Popović, Teresa Vidal-Calleja, Gregory Hitz, Jen Jen Chung, Inkyu Sa, Roland Siegwart, and Juan Nieto. An Informative Path Planning Framework for UAV-based Terrain Monitoring. *Autonomous Robots*, 44 (6):889–911, 2020.

[127] Marija Popović, Joshua Ott, Julius Rückin, and Mykel J. Kochenderfer. Learning-based Methods for Adaptive Informative Path Planning. *Journal on Robotics and Autonomous Systems (RAS)*, 179:104727, 2024.

[128] Janis Postels, Mattia Segu, Tao Sun, Luca Daniel Sieber, Luc Van Gool, Fisher Yu, and Federico Tombari. On the Practicality of Deterministic Epistemic Uncertainty. In *Proc. of the Intl. Conf. on Machine Learning (ICML)*, 2022.

[129] Steven Reece and Stephen Roberts. An introduction to Gaussian Processes for the Kalman Filter Expert. In *Intl. Conf. on Information Fusion*, 2010.

[130] Russell Reinhart, Tung Dang, Emily Hand, Christos Papachristos, and Kostas Alexis. Learning-based Path Planning for Autonomous Exploration of Subterranean Environments. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2020.

[131] Eduardo Romera, José M. Alvarez, Luis M. Bergasa, and Roberto Arroyo. ERFNet: Efficient Residual Factorized ConvNet for Real-Time Semantic

Segmentation. *IEEE Trans. on Intelligent Transportation Systems (T-ITS)*, 19(1):263–272, 2018.

[132] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Proc. of the Int. Conf. on Medical Image Computing and Computer-Assisted Intervention*, 2015.

[133] Franz Rottensteiner, Gunho Sohn, Jaewook Jung, Markus Gerke, Caroline Baillard, Sebastien Benitez, and Uwe Breitkopf. The ISPRS Benchmark on Urban Object Classification and 3D Building Reconstruction. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 1(1):293–298, 2012.

[134] Julius Rückin, Liren Jin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Informative Path Planning for Active Learning in Aerial Semantic Mapping. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.

[135] Julius Rückin, Liren Jin, and Marija Popović. Adaptive Informative Path Planning Using Deep Reinforcement Learning for UAV-based Active Sensing. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[136] Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. An Informative Path Planning Framework for Active Learning in UAV-Based Semantic Mapping. *IEEE Trans. on Robotics (TRO)*, 39(6):4279–4296, 2023.

[137] Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Active Learning of Robot Vision Using Adaptive Path Planning. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS) Workshop on Label Efficient Learning Paradigms for Autonomy at Scale*, 2024.

[138] Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Semi-Supervised Active Learning for Semantic Segmentation in Unknown Environments Using Informative Path Planning. *IEEE Robotics and Automation Letters (RA-L)*, 9(3):2662–2669, 2024.

[139] Julius Rückin, David Morilla-Cabello, Cyrill Stachniss, Eduardo Montijano, and Marija Popović. Towards Map-Agnostic Policies for Adaptive Informative Path Planning. *IEEE Robotics and Automation Letters (RA-L)*, 10(5): 5114–5121, 2025.

[140] Inkyu Sa, Zetao Chen, Marija Popović, Raghav Khanna, Frank Liebisch, Juan Nieto, and Roland Siegwart. Weednet: Dense Semantic Weed Classification Using Multispectral Images and MAV for Smart Farming. *IEEE Robotics and Automation Letters (RA-L)*, 3(1):588–595, 2018.

[141] Manish Saroya, Graeme Best, and Geoffrey A. Hollinger. Online Exploration of Tunnel Networks Leveraging Topological CNN-based World Predictions. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[142] Lukas Schmid, Michael Pantic, Raghav Khanna, Lionel Ott, Roland Siegwart, and Juan Nieto. An Efficient Sampling-based Method for Online Informative Path Planning in Unknown Environments. *IEEE Robotics and Automation Letters (RA-L)*, 5(2):1500–1507, 2020.

[143] Lukas Schmid, Jeffrey Delmerico, Johannes L. Schönberger, Juan Nieto, Marc Pollefeys, Roland Siegwart, and Cesar Cadena. Panoptic Multi-TSDFs: A Flexible Representation for Online Multi-Resolution Volumetric Mapping and Long-Term Dynamic Scene Consistency. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

[144] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model. *Nature*, 588(7839):604–609, 2020.

[145] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-Dimensional Continuous Control Using Generalized Advantage Estimation. *arXiv preprint, arXiv:1506.02438*, 2015.

[146] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms. *arXiv preprint, arXiv:1707.06347*, 2017.

[147] Ozan Sener and Silvio Savarese. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *Proc. of the Int. Conf. on Learning Representations (ICLR)*, 2018.

[148] Burr Settles. Active Learning Literature Survey. Technical report, Computer Sciences Department of the University of Wisconsin–Madison, 2009.

[149] Gyungin Shin, Weidi Xie, and Samuel Albanie. All You Need Are a Few Pixels: Semantic Segmentation with PixelPick. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[150] Rakesh Shrestha, Fei-Peng Tian, Wei Feng, Ping Tan, and Richard Vaughan. Learned Map Prediction for Enhanced Mobile Robot Exploration. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2019.

[151] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor Segmentation and Support Inference from RGBD Images. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2012.

[152] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the Game of Go with Deep Neural Networks and Tree Search. *Nature*, 529(7587):484–489, 2016.

[153] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A General Reinforcement Learning Algorithm that Masters Chess, Shogi, and Go through Self-Play. *Science*, 362(6419):1140–1144, 2018.

[154] Robert Sim and Nicholas Roy. Global A-Optimal Robot Exploration in SLAM. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2005.

[155] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational Adversarial Active Learning. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2019.

[156] Leslie N. Smith. A Disciplined Approach to Neural Network Hyper-Parameters: Part 1–Learning Rate, Batch Size, Momentum, and Weight Decay. *arXiv preprint, arXiv:1803.09820*, 2018.

[157] Yunlong Song, Selim Naji, Elia Kaufmann, Antonio Loquercio, and Davide Scaramuzza. Flightmare: A Flexible Quadrotor Simulator. In *Proc. of the Conf. on Robot Learning (CoRL)*, 2021.

[158] Yunlong Song, Angel Romero, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. Reaching the Limit in Autonomous Racing: Optimal Control versus Reinforcement Learning. *Science Robotics*, 8(82):eadg1462, 2023.

[159] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal on Machine Learning Research (JMLR)*, 15:1929–1958, 2014.

[160] Zachary N. Sunberg and Mykel J. Kochenderfer. Online Algorithms for POMDPs with Continuous State, Action, and Observation Spaces. In *Proc. of the Int. Conf. on Automated Planning and Scheduling (ICAPS)*, 2018.

[161] Yoonchang Sung, Jnaneshwar Das, and Pratap Tokekar. Decision-Theoretic Approaches for Robotic Environmental Monitoring - A Survey. *arXiv preprint, arXiv:2308.02698*, 2023.

[162] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT press, 2018.

[163] Chee S. Tan, Rosmiwati Mohd-Mokhtar, and Mohd R. Arshad. A Comprehensive Review of Coverage Path Planning in Robotics Using Classical and Heuristic Algorithms. *IEEE Access*, 9:119310–119342, 2021.

[164] Yuezhan Tao, Yuwei Wu, Beiming Li, Fernando Cladera, Alex Zhou, Dinesh Thakur, and Vijay Kumar. SEER: Safe Efficient Exploration for Aerial Robots using Learning to Predict Information Gain. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[165] Simon Tong and Daphne Koller. Support Vector Machine Active Learning with Applications to Text Classification. *Journal on Machine Learning Research (JMLR)*, 2(Nov):45–66, 2001.

[166] Devis Tuia, Frédéric Ratle, Fabio Pacifici, Mikhail F. Kanevski, and William J. Emery. Active Learning Methods for Remote Sensing Image Classification. *IEEE Trans. on Geoscience and Remote Sensing*, 47(7): 2218–2232, 2009. doi: 10.1109/TGRS.2008.2010404.

[167] Apoorva Vashisth, Julius Rückin, Federico Magistri, Cyrill Stachniss, and Marija Popović. Deep Reinforcement Learning with Dynamic Graphs for Adaptive Informative Path Planning. *IEEE Robotics and Automation Letters (RA-L)*, 9(9):7747–7754, 2024.

[168] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2017.

[169] Alberto Viseras and Ricardo Garcia. DeepIG: Multi-Robot Information Gathering with Deep Reinforcement Learning. *IEEE Robotics and Automation Letters (RA-L)*, 4(3):3059–3066, 2019.

[170] Alberto Viseras, Michael Meissner, and Juan Marchal. Wildfire Front Monitoring with Multiple UAVs Using Deep Q-Learning. *IEEE Access*, pages 1–1, 2021.

[171] Kelen C.T. Vivaldini, Thiago H. Martinelli, Vitor C. Guizilini, Jefferson R. Souza, Matheus D. Oliveira, Fabio T. Ramos, and Denis F. Wolf. UAV Route Planning for Active Disease Classification. *Autonomous Robots*, 43 (5):1137–1153, 2019.

[172] Stavros G. Vougioukas. Agricultural Robotics. *Annual Review of Control, Robotics, and Autonomous Systems*, 2(1):365–392, 2019.

[173] Zheng Wang and Jieping Ye. Querying Discriminative and Representative Samples for Batch Mode Active Learning. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 9(3):1–23, 2015.

[174] Yongyong Wei and Rong Zheng. Informative Path Planning for Mobile Sensing with Reinforcement Learning. In *Proc. of the IEEE Conf. on Computer Communications*, 2020.

[175] Jonas Westheider, Julius Rückin, and Marija Popović. Multi-UAV Adaptive Path Planning Using Deep Reinforcement Learning. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[176] Jan Weyler, Federico Magistri, Elias Marks, Yue Linn Chong, Matteo Sodano, Gianmarco Roggiolani, Nived Chebrolu, Cyrill Stachniss, and Jens Behley. PhenoBench: A Large Dataset and Benchmarks for Semantic Image Interpretation in the Agricultural Domain. *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, pages 1–12, 2024.

[177] Christopher K.I. Williams and Carl E. Rasmussen. *Gaussian Processes for Machine Learning*, volume 2. MIT Press, 2006.

[178] Ronald J. Williams. Simple Statistical Gradient-following Algorithms for Connectionist Reinforcement Learning. *Machine learning*, 8:229–256, 1992.

[179] Sjaak Wolfert, Lan Ge, Cor Verdouw, and Marc-Jeroen Bogaardt. Big Data in Smart Farming - A review. *Agricultural Systems*, 153:69–80, 2017.

[180] David J. Wu. Accelerating Self-Play Learning in Go. *arXiv preprint, arXiv:1902.10565*, 2019.

[181] Binhui Xie, Longhui Yuan, Shuang Li, Chi Harold Liu, and Xinjing Cheng. Towards Fewer Annotations: Active Learning via Region Impurity and Prediction Uncertainty for Domain Adaptive Semantic Segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[182] Brian Yamauchi. A Frontier-based Approach for Autonomous Exploration. In *Proc. of the IEEE Intl. Symp. on Computer Intelligence in Robotics and Automation (CIRA)*, 1997.

[183] Lin Yang, Yizhe Zhang, Jianxu Chen, Siyuan Zhang, and Danny Z. Chen. Suggestive Annotation: A Deep Active Learning Framework for Biomedical Image Segmentation. In *Proc. of the Int. Conf. on Medical Image Computing and Computer-Assisted Intervention*, 2017.

[184] Pengzhi Yang, Yuhan Liu, Shumon Koga, Arash Asgharivaskasi, and Nikolay Atanasov. Learning Continuous Control Policies for Information-Theoretic Active Perception. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2023.

[185] Tianze Yang, Yuhong Cao, and Guillaume Sartoretti. Intent-based Deep Reinforcement Learning for Multi-Agent Informative Path Planning. In *Proc. of the Intl. Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, 2023.

[186] Yasuyoshi Yokokohji. The Use of Robots to Respond to Nuclear Accidents: Applying the Lessons of the Past to the Fukushima Daiichi Nuclear Power Station. *Annual Review of Control, Robotics, and Autonomous Systems*, 4 (1):681–710, 2021.

[187] Tobias Zaenker, Julius Rückin, Rohit Menon, Marija Popović, and Maren Bennewitz. Graph-based View Motion Planning for Fruit Detection. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[188] Dimitris Zermas, Izzat Izzat, and Nikolaos Papanikolopoulos. Fast Segmentation of 3D Point Clouds: A Paradigm on Lidar Data for Autonomous Vehicle Applications. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2017.

[189] Yifan Zhang, Bingyi Kang, Bryan Hooi, Shuicheng Yan, and Jiashi Feng. Deep Long-Tailed Learning: A Survey. *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, 45(9):10795–10816, 2023.

[190] Fedor Zhdanov. Diverse Mini-Batch Active Learning. *arXiv preprint, arXiv:1901.05954*, 2019.

[191] René Zurbrügg, Hermann Blum, Cesar Cadena, Roland Siegwart, and Lukas Schmid. Embodied Active Domain Adaptation for Semantic Segmentation via Informative Path Planning. *IEEE Robotics and Automation Letters (RA-L)*, 7(4):8691–8698, 2022.

[192] Elchanan Zwecher, Eran Iceland, Sean R. Levy, Shmuel Y. Hayoun, Oren Gal, and Ariel Barel. Integrating Deep Reinforcement and Supervised Learning to Expedite Indoor Mapping. In *Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA)*, 2022.

List of Figures

List of Tables

List of Algorithms