UNIVERSITÄT BONN

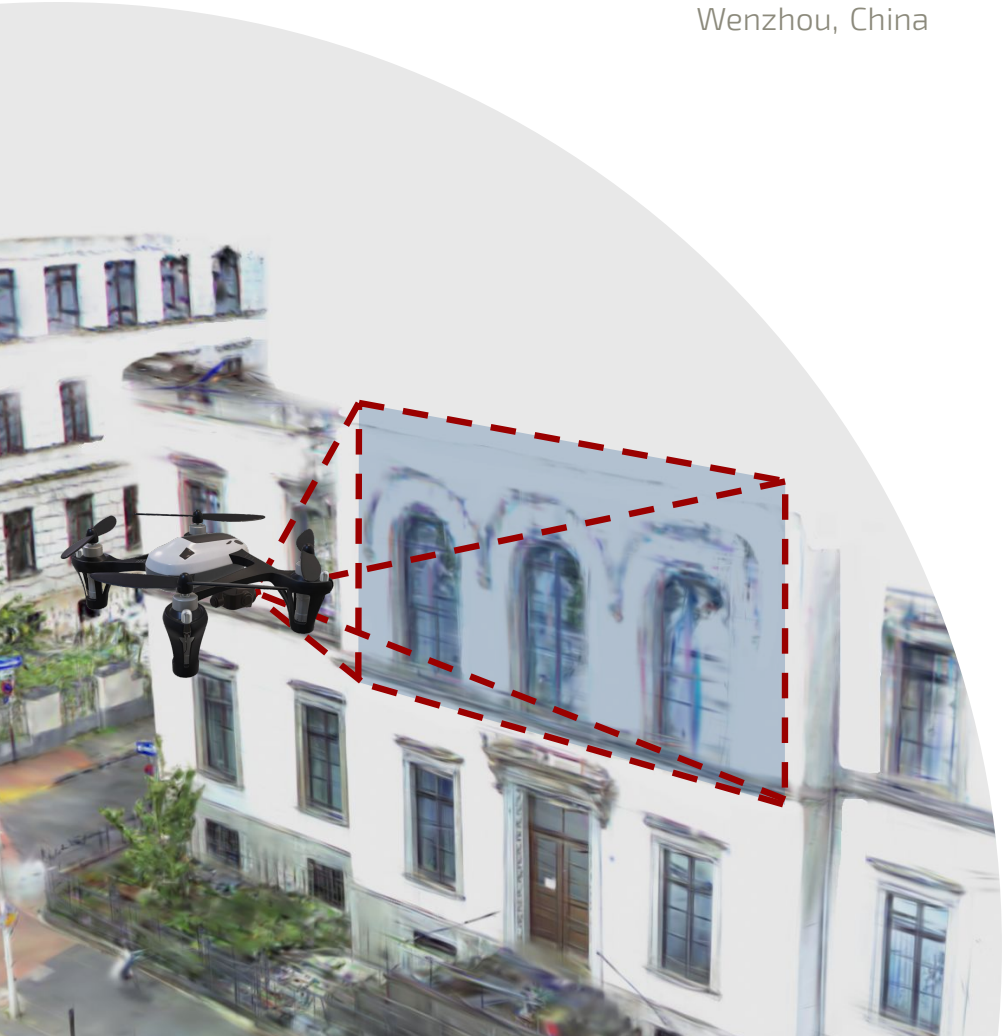
Dissertation
zur Erlangung des Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)
Agrar-, Ernährungs- und Ingenieurwissenschaftliche Fakultät
der Rheinischen Friedrich-Wilhelms-Universität Bonn
Institut für Geodäsie und Geoinformation

Active Perception for Learning-Based Robot Mapping

von

Liren Jin

aus
Wenzhou, China



Bonn 2026

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

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

2. Korreferent:
Prof. Dr. Hermann Blum, University of Bonn, Germany

Tag der mündlichen Prüfung: 14.01.2026

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

Zusammenfassung

Autonome Roboter müssen ihre Umgebung wahrnehmen und verstehen, um verschiedenste Aufgaben erfolgreich planen und ausführen zu können. Ein grundlegender Aspekt dieser Wahrnehmungsfähigkeit besteht darin, die Aufnahmepositionen der Sensoren aktiv anzusteuern, um die Umgebung zu erkunden und aufgabenrelevante, informative Messungen zu erfassen. Im Gegensatz zur passiven Wahrnehmung, die vordefinierte Pfade oder feste Heuristiken zur Exploration folgt, und zur externen Überwachung, die arbeitsintensive menschliche Anleitung erfordert, beinhaltet aktive Wahrnehmung autonome Entscheidungsprozesse, bei denen der Roboter basierend auf seinem aktuellen Wissensstand über die Umgebung die aussichtsreichsten Aufnahmepositionen für das Sammeln von Messungen auswählt. Der entscheidende Schritt in diesem Prozess ist die Observationsplanung, die es dem Roboter ermöglicht, Aufnahmepositionen auszuwählen, die den erwarteten Nutzen der erfassten Messungen maximieren. Diese Fähigkeit ist besonders relevant in unbekannten Umgebungen, in denen kein Vorwissen zur Unterstützung der Observationsplanung zur Verfügung steht, und ihre Online-Anpassung kann die Leistung bei Aufgaben wie Lokalisierung, Objekterkennung und Kartierung verbessern.

In dieser Dissertation konzentrieren wir uns auf die Roboterkartierung, bei der Roboter mit bordeigenen Sensoren eingesetzt werden, um räumliche Repräsentationen ihrer Umgebung zu erstellen. Insbesondere untersuchen wir die autonome Kartierung in unbekannten Umgebungen mittels Integration aktiver Wahrnehmungsstrategien. Unser Ziel ist es, Robotern zu ermöglichen, mithilfe von Sensormessungen präzise räumliche Repräsentationen aktiv zu erstellen. Während frühere Arbeiten bereits aktive Wahrnehmung für Roboterkartierung untersucht haben, konzentrieren sich viele bestehende Ansätze nicht darauf, feingranulare Details der Umgebung zu bewahren – Details, die für Anwendungen mit hohen Anforderungen an Modelltreue entscheidend sind, wie etwa die Inspektion von Infrastrukturen und die Erstellung digitaler Zwillinge. Dies liegt hauptsächlich an der Verwendung konventioneller, diskreter Kartenrepräsentationen, die oft mit Informationsverlust während des Kartierungsprozesses einhergehen.

Wir lösen diese Herausforderung durch den Einsatz lernbasierter Kartierungs-

techniken, die die Umgebung kontinuierlich darstellen können. Der Hauptbeitrag dieser Dissertation liegt in der Entwicklung aktiver Wahrnehmungssysteme, die lernbasierte Kartierungsmethoden einsetzen. Wir untersuchen Gauss-Prozesse, image-based neural rendering, semantic neural radiance fields und Gaussian splatting, um eine autonome, hochpräzise Roboterkartierung zu realisieren. Im Zentrum unserer Systeme steht die Anpassung der Kartenrepräsentationen sowie die Entwicklung von Nutzenfunktionen, die den erwarteten Nutzen möglicher Aufnahmepositionen in Bezug auf spezifische Kartierungsziele bewerten, wie z. B. Reduzierung von Kartenunsicherheit oder die Steigerung der Rekonstruktionsgenauigkeit, um die Integration von aktiver Wahrnehmung zu ermöglichen. Aufgrund der unterschiedlichen Eigenschaften dieser Kartierungstechniken entwickeln wir für jede Methode maßgeschneiderte aktive Wahrnehmungsstrategien, um die Observationsplanung mit der zugrunde liegenden Kartenstruktur abzustimmen. Zur Validierung unserer Beiträge evaluieren wir die vorgeschlagenen Methoden sowohl in Simulationen als auch in realen Szenarien und zeigen ihre Vorteile hinsichtlich Effizienz und Qualität bei autonomen Kartierungsaufgaben.

Des Weiteren zeigt diese Dissertation die Wirksamkeit der aktiven Wahrnehmung für lernbasierte Roboterkartierung. Durch die Verknüpfung adaptiver Observationsplanung mit lernbasierten Kartierungstechniken leistet unsere Arbeit einen wichtigen Beitrag zur aktiven Wahrnehmung für Roboterkartierung und ermöglicht eine effizientere sowie genauere Modellierung unbekannter Umgebungen. Sämtliche in dieser Dissertation vorgestellten Methoden wurden in begutachteten Konferenzbeiträgen und Zeitschriftenartikeln veröffentlicht und leisten somit einen wissenschaftlich fundierten Beitrag zum Forschungsfeld. Um Reproduzierbarkeit zu fördern und zukünftige Entwicklungen zu unterstützen, wurde der zugehörige Quellcode in öffentlich zugänglichen Repositorien bereitgestellt.

Abstract

AUTONOMOUS robots need to perceive and understand their environment in order to plan and carry out tasks. A fundamental aspect of this perception capability is the active control of onboard sensor viewpoints to explore the surrounding environment and acquire informative measurements relevant to the task at hand. Unlike passive perception, which follows predefined path patterns or fixed heuristics for exploration, and external supervision, which requires labor-intensive human guidance, active perception involves autonomous decision-making to determine the most valuable viewpoints for collecting measurements based on the robot's current knowledge of the environment. The key in the process is the view planning step, which enables the robot to select viewpoints that maximize the expected usefulness of the acquired measurements. This capability is relevant in unknown environments, where prior knowledge is unavailable to inform view planning, and its online adaptation can enhance performance for tasks such as localization, object detection, and mapping.

In this thesis, we focus on the task of robot mapping, using robots equipped with onboard sensors to construct spatial representations of their environments. Specifically, we investigate autonomous mapping in unknown environments by integrating active perception strategies. Our goal is to enable robots to actively build accurate spatial representations using sensor measurements. While previous work has studied active perception for robot mapping, many existing approaches do not focus on preserving fine-grained details of the environment, which are crucial for tasks requiring high-fidelity environmental models, including infrastructure inspection and digital twin generation. This largely stems from the use of conventional, discrete map representations, which lead to information loss during the mapping process.

We address this challenge by leveraging learning-based mapping techniques capable of representing the environment in a continuous manner. The main contribution of this thesis is the development of active perception strategies with such mapping techniques. We explore Gaussian processes, image-based neural rendering, semantic neural radiance fields, and Gaussian splatting to achieve au-

tonomous, high-fidelity robot mapping. At the core of our approach lies the adaptation of map representations and the design of utility formulations that assess the expected usefulness of candidate viewpoints with respect to specific mapping objectives, such as reducing map uncertainty or enhancing reconstruction fidelity, thereby enabling active perception. Due to the varying characteristics of these mapping techniques, we develop tailored active perception strategies for each method to align the view planning module with the underlying map representation. To validate our contributions, we evaluate the proposed methods in simulation and real-world scenarios, demonstrating their strengths in improving mapping efficiency and quality for autonomous mapping tasks.

Overall, this thesis highlights the effectiveness of active perception for learning-based robot mapping. By coupling view planning with learning-based mapping techniques, our work takes an important step forward in the field of active perception for robot mapping, contributing to more efficient and accurate environmental modeling in unknown environments. All methods presented in this thesis have been published in peer-reviewed conference papers and journal articles, underscoring their scientific contribution to the field. To support reproducibility and further research, the corresponding source code has been made publicly available in open-access repositories.

Acknowledgements

DOING a PhD was never part of my original plan when I first began my Master's program in Germany in the fall of 2016. Yet life often unfolds in unexpected ways, and looking back, I am deeply grateful for the path that brought me here. I am proud to have found the courage to make this decision and the determination to embark on this journey.

First and foremost, I owe my deepest gratitude to my supervisor, Marija Popović — Masha. I still remember the joy and excitement when I received your call offering me the chance to pursue a PhD research in Bonn. Over these years, you have been a constant source of support, always open for discussion, and generous in giving me the freedom to explore the ideas that truly inspired me. Your trust and encouragement shaped not only my research but also my growth as a person. I would like to express my sincere thanks to my co-supervisor, Cyrill Stachniss. Your open-mindedness, determination, and scientific excellence set an example for me to aspire to. It has been a privilege to learn from your guidance and leadership. I am also grateful to Hermann Blum for dedicating your time to reviewing my thesis. Your effort is greatly appreciated.

A heartfelt thanks to Julius Rückin, my officemate throughout this PhD journey. Our countless discussions, both serious and lighthearted, made research far more interesting and fulfilling. To Xieyuanli (Rhiney) Chen, who offered me so much valuable advice when I sometimes felt lost at the start of my PhD. Your kindness and wisdom meant more than I can express. To Jens Behley, whose knowledge and encouragement always pushed me forward, I appreciate those long conversations that inspired me a lot. It is a great fortune for me to have a small Chinese community here in the lab with Haofei Kuang, Xingguang (Starry) Zhong, and Yue Pan. Thank you for the joyful moments we have shared, both in research and leisure. And to my other wonderful colleagues who have been part of my daily PhD life: Jonas Westheider, Perrine Aguiar, Birgit Klein, Thomas Läbe, Kirsten Sadler, Yue (Linn) Chong, Saurabh Gupta, Luca Lobefaro, Meher Malladi, Rodrigo Marcuzzi, Elias Marks, Benedikt Mersch, Lucas Nunes, Gianmarco Roggiolani, Matteo Sodano, Niklas Trekel, Louis Wiesmann, Jan Weyler, Federico Magistri, Tiziano Guadagnino, and Matthias Zeller. Thank you all for

creating such a warm and inspiring atmosphere. Working alongside you has been one of the most rewarding parts of this experience.

Beyond the lab, I am grateful to Maren Bennewitz, Sicong Pan, Xuying Huang, Teresa Vidal-Calleja, Stefan Kiss, and Hao Hu for the fruitful collaborations and stimulating discussions. I also want to thank the members of the cluster office, Franziska Kübel, Sonja de Vries, Katharina Monaco, Nora Berning, and all the others for your continuous support behind the scenes.

Special thanks to Xing Li, who became a close friend during our internship together and encouraged me to pursue a PhD. Without your inspiration, my journey might have been very different. To my best friends Yang Zhang, Daojing Lin, and Han Wu. Though we are scattered across the world, our friendship has never faded. Every conversation with you fills me with strength and energy.

Last but certainly not least, I thank my parents, Wanli and Songlan. Nothing would have been possible without your unwavering love and support from day one. Your guidance and teachings have shaped who I am today. I am grateful to my big family, always cheering for me from afar, and to all the people who, in one way or another, supported me throughout this journey. A warm thanks as well to my cat, Dumm, for always waiting for me to come home and for using your superpower to relieve all my stress and frustration.

This PhD was not just a chapter of research; it has been a journey of growth, humility, resilience, and gratitude.

This work has been fully 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

Acronyms

AUSE area under sparsification error

GP Gaussian process

GS Gaussian splatting

IoU intersection over union

MLP multi-layer perceptron

NBV next best view

NeRF neural radiance field

PSNR peak signal-to-noise ratio

RMSE root mean square error

SLAM simultaneous localization and mapping

SRCC Spearman's rank correlation coefficient

SSIM structural similarity index

UAV unmanned aerial vehicle

Chapter 1

Introduction

Robots are becoming increasingly popular in various fields, including agricultural, industrial, and household service applications. To autonomously operate in the real world, robots must be able to perceive their environment using sensors, such as cameras and LiDARs, and make informed decisions based on the information contained in the acquired measurements. This capability becomes particularly crucial in unknown environments, where a robot has no prior knowledge of the scene and thus heavily relies on online perception.

Consider how humans explore a new environment: they actively look around, seek new viewpoints to avoid occlusions, and approach certain regions to gather detailed information, such as the attributes of objects or the spatial layout of the scene. The perceived visual information forms the basis for decision-making, guiding subsequent movements and interactions with objects in the environment. Such human-like exploration behavior naturally embodies the concept of active perception [4], in which the perception process of where to collect measurements is actively controlled to acquire useful information relevant to the tasks.

While active perception is natural and intuitive for humans, traditional robotic systems have adopted a different strategy in practice. In many applications, robots primarily serve as a platform for carrying sensors, where the perception process is conducted manually or passively. It either relies on external control, e.g., from human operators, or simply follows predefined path patterns or heuristics, for collecting measurements in unknown environments. This strategy is simple to implement but often lacks informative feedback from the perception process itself and cannot automatically adjust its sensor viewpoints based on the information in the acquired measurements, resulting in inherent limitations. For instance, external control typically requires human-in-the-loop operation, which can be labor-intensive, delayed, and heavily dependent on the operators' experience. This also bottlenecks the scalability of robotic systems, as it is difficult to

Figure 1.1: Active perception enables a robot to actively decide where to collect informative measurements. This is relevant for achieving robot autonomy in unknown environments, as no prior knowledge is available and robots need to adapt their perception strategies on the fly based on the information in the acquired measurements. For instance, to inspect or pick fruits, a robot arm needs to actively reposition its onboard camera to better perceive the target fruit and avoid occlusions, such as by following a path indicated by the dashed line. Image adapted from Federico Magistri.

deploy a large number of robots in parallel solely under human supervision. On the other hand, designing a perception strategy in the absence of prior knowledge of a specific environment is often infeasible, while blindly following fixed path patterns or heuristics without online adaptation can lead to inferior task performance, as such approaches fail to leverage information already obtained for deciding where to collect measurements next. In contrast, robot autonomy demands onboard decision-making for robots operating in unknown environments, free from external interventions or rigid, predefined plans.

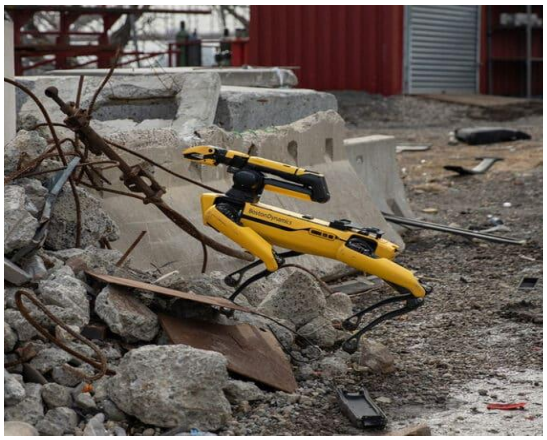
By tightly coupling perception and sensor control, active perception builds up a closed-loop system, allowing for adaptive perception strategies during online missions. For example, a robot may actively adjust its sensor viewpoints based on current measurements to avoid occlusions or to focus on detected objects of interest, as illustrated in Figure 1.1. This closed-loop behavior facilitates more targeted perception by directing attention to specific regions in the environment, thereby increasing the efficiency of measurement acquisition.

Active perception therefore plays an important role in enhancing robot autonomy in unknown environments [4], and it has actually been widely applied

in tasks, such as object recognition [32, 129, 205], localization [19, 40, 49, 88, 165], semantic scene understanding [146, 214, 216, 222], and mapping [30, 38, 164, 204]. In the context of object recognition, a robot selects viewpoints that maximize the visibility of an object or minimize the ambiguity, leading to improved recognition performance. For localization, a robot plans viewpoints to target texture-rich areas to minimize uncertainty in its pose estimate or actively search for loop closures to reduce accumulated drift, achieving more accurate localization. For semantic scene understanding, a robot gathers information about object relationships or semantic attributes, enhancing its holistic interpretation of the scene. This thesis focuses specifically on active perception for robot mapping, a concept we also use interchangeably with the term active reconstruction in the following chapters. In this context, the integration of perception and sensor control is fundamental to enabling robots to autonomously and efficiently construct map representations of previously unknown environments.

Robot mapping is a fundamental task in robotics, where robots create spatial representations of the environment, identifying what the world looks like, where things are located, and potentially the attributes of those objects. This spatial understanding supports a wide range of robotic applications, including search and rescue, infrastructure inspection, agricultural monitoring, and household service, as shown in Figure 1.2. For example, robots can be used to generate 3D maps of collapsed buildings, providing structural information to support rescue operations. In infrastructure inspection, where human access can be difficult or even hazardous, robots can build a map to help identify potential issues in structures, such as bridges or factory facilities, supporting preventive maintenance. In precision agriculture, field monitoring using robots is often more efficient than manual inspection and also provides higher flexibility compared to static sensor networks. An accurate map of the field helps farmers to optimize crop management and resource allocation. In household service or warehouse automation, robot manipulation requires accurate spatial modeling of the target objects, which is necessary for planning and executing physical interactions, such as grasping and assembly. In this thesis, we tackle the problem of autonomously generating accurate environmental models using robots equipped with onboard sensors, without any human supervision, to exploit the potential of robot autonomy for mapping.

When deployed in real-world scenarios, autonomous robot mapping systems often operate under mission constraints, such as limited operation time or travel distance, which necessitate effective perception strategies for collecting informative measurements for map updates. Passive perception for robot mapping in unknown environments may lead to incomplete, redundant, or poorly targeted measurements, degrading both mapping efficiency and map quality. This limitation arises from fixed perception strategies in passive perception, which cannot



(a) Search and rescue

(b) Infrastructure inspection

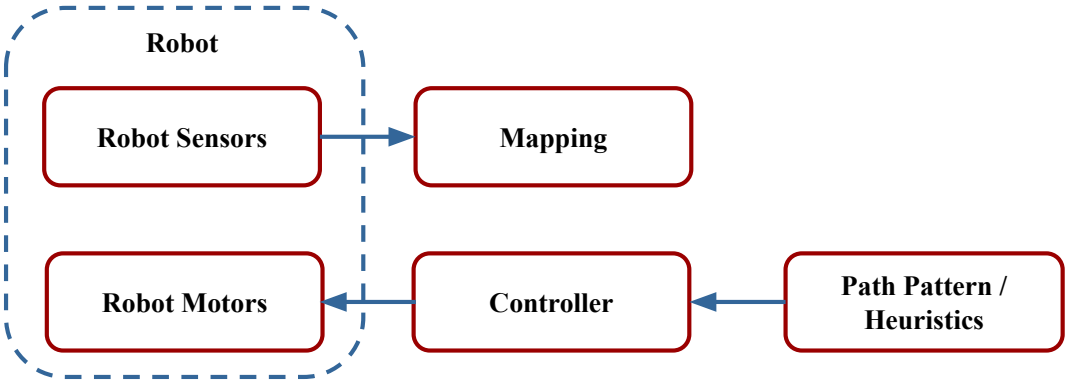
(c) Agricultural monitoring

(d) Household service

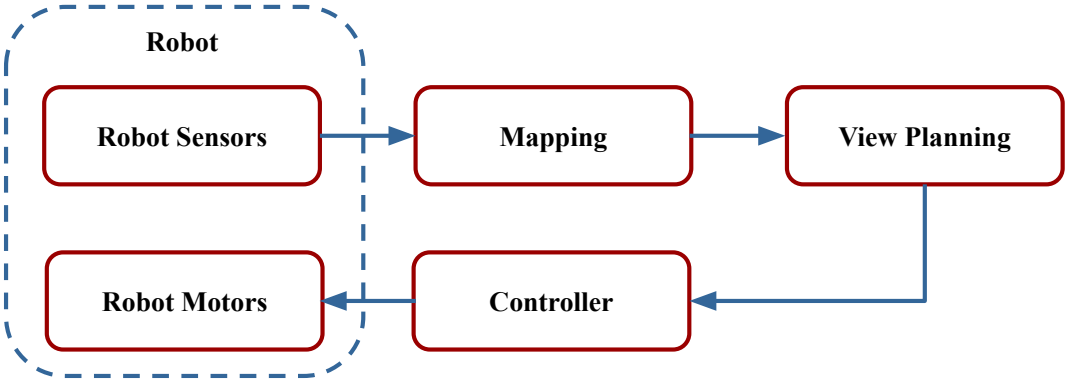
Figure 1.2: Examples of different application scenarios for robot mapping. Accurate mapping techniques are important for applications such as search and rescue, infrastructure inspection, agricultural monitoring, and household service. Compared to static sensor networks or manual inspection, robot mapping offers significant advantages in terms of accessibility, flexibility, efficiency, and autonomy in these applications. Images from Boston Dynamics, ExRobotics, DJI, and ACRV.

account for current map states during online missions. As a consequence, passive perception proves inefficient and falls short of fully optimizing the mapping objectives under mission constraints.

We compare the frameworks of passive and active perception for robot mapping in Figure 1.3 to highlight the differences in these two paradigms. At the core of active perception for robot mapping is the view planning problem, which involves selecting informative viewpoints to acquire new measurements that improve the map representation. By explicitly taking into account the current map state and mapping objectives for view planning, active perception methods close



(a) Passive perception for robot mapping



(b) Active perception for robot mapping

Figure 1.3: A comparison between the general frameworks of passive and active perception for robot mapping. (a) In passive perception, the robot follows a predefined path or non-adaptive heuristic to collect measurements for mapping, which may result in redundant or incomplete measurements of the environment, leading to inefficient or low-quality mapping. (b) In contrast, active perception enables the robot to actively select informative viewpoints based on the current map state, directly optimizing specific mapping objectives, such as full coverage or high reconstruction accuracy. This closed-loop setup enhances the robot autonomy and leads to more efficient and accurate mapping in unknown environments.

the loop between perception and sensor control, overcoming the limitations of passive perception and enabling more accurate and efficient mapping.

Active perception for robot mapping aims to improve specific mapping objectives in unknown environments, such as maximizing scene coverage, reducing uncertainty, or enhancing overall map quality, under mission constraints. Common strategies in this domain follow the next best view (NBV) paradigm [132, 202], which often involves viewpoint generation and evaluation to greedily decide where to collect next measurements for map updates. In this process, viewpoint generation is responsible for proposing possible candidate viewpoints, while viewpoint evaluation estimates the expected utility values of measurements at these

viewpoints, i.e., how much a potential measurement at these viewpoints would contribute toward improving the mapping objectives. The generated viewpoints are then ranked based on their expected utility, and the robot selects the best viewpoint to move to for taking new measurements [16]. However, acquiring the utility values of candidate viewpoints can be non-trivial. It typically requires either computationally intensive simulation of measurements at those viewpoints to estimate the resulting map posterior, or evaluations of current map quality from those viewpoints in the absence of ground truth. These requirements pose significant challenges for utility evaluation often necessary for view planning in unknown environments. Moreover, the diversity of map representations and task-specific mapping objectives calls for customized utility functions, and in many cases, even modifications to the map representations themselves to support active perception. The works presented in this thesis also follow the NBV paradigm and aim to tackle these challenges, enabling active perception for robot mapping.

1.1 Point of Departure

At the outset of this PhD research, active perception for robot mapping was already established in the robotics community, with a variety of methods developed for different mapping objectives and map representations [96]. However, the majority of these methods are primarily designed with a focus on spatial exploration to improve the map coverage or coarse geometric modeling, while often overlooking fine-grained details in the environment.

Commonly used map representations in previous approaches are conventional robotic maps such as voxel grids [11,152,219], point clouds [28,31,210], or surface meshes [160, 161]. While effective in many applications, these representations often suffer from discretization artifacts, and the fixed shape primitives of the map cannot adapt to the varying levels of detail required in different areas of the environment. This inherently limits their ability to preserve fine-grained details that are crucial for tasks requiring high-fidelity environmental models, such as object-level manipulation, detailed inspection, or photorealistic rendering. For instance, space discretization in voxel grids leads to information loss in areas with fine structural details unless extremely high resolutions are used, which is at the cost of memory and computational resources. Point clouds, on the other hand, provide a more flexible representation but often lack the density and connectivity needed to accurately model complex surface geometry and texture. Surface meshes show inflexibility for online incremental mapping due to their fixed surface pattern and usually require post-processing to acquire the final meshes.

The gap between the existing active perception approaches and the need for high-fidelity scene modeling motivates the direction of this thesis. We aim to

develop active perception methods that leverage more advanced map representations capable of preserving environmental details for autonomous robot mapping.

With the growing demands for more accurate and detailed mapping techniques, learning-based map representations that can continuously capture environmental attributes have become increasingly popular in robotics applications [102, 104, 117, 168]. Techniques such as Gaussian process (GP) models and radiance field representations represented by neural radiance field (NeRF), and more recently, Gaussian splatting (GS), have been adopted for robot mapping. These methods have gained enormous attention for their ability to create highly accurate environmental models by directly learning from measurement data, surpassing the representation capabilities of traditional mapping approaches. For example, GPs can be used to probabilistically model the spatial distribution of physical phenomena, e.g., field temperature or gas density, allowing for continuous mapping and uncertainty estimation [65, 117, 167]. Radiance fields, on the other hand, represent complex scene geometry and texture by training a radiance field of the environment from dense measurements, enabling photorealistic rendering from novel viewpoints [69, 104, 144, 172].

Such learning-based representations offer the promise of high-fidelity mapping; however, they also introduce new challenges for integrating active perception. For instance, GPs are inherently capable of modeling uncertainty in the map, which is particularly valuable in active perception, allowing the robot to prioritize areas of high uncertainty and thereby improve the efficiency of exploration and mapping. Therefore, GP-based methods have been widely used in active perception for robot mapping in different scenarios. Nonetheless, GPs require careful consideration of the kernel functions to capture spatial correlations present in the mapping target. They also suffer from high computational costs when dealing with dense measurements, which can limit their applicability for online incremental robot mapping. Recently emerging techniques such as NeRFs and GS have demonstrated great potential for photorealistic robot mapping. Yet, research in this area has primarily focused on improving reconstruction quality in offline settings, where all scene measurements are precollected and maps are constructed post hoc, rather than incrementally during an online mission. Their integration into active perception remains largely unexplored. A key question lies in the fact that these representations do not provide explicit uncertainty estimates as GPs do, and require additional mechanisms to evaluate the utility of candidate viewpoints for view planning in active perception.

These challenges give rise to a core research question in this thesis: Can we effectively integrate active perception strategies with these learning-based map representations, achieving high-fidelity mapping results in a computationally efficient and autonomous manner in unknown environments?

1.2 Main Contributions

This thesis aims to jointly consider active perception and learning-based robot mapping techniques. We focus on two main aspects of active perception for robot mapping: (1) the design of utility formulations tailored to different learning-based map representations and mapping objectives; and (2) the adaptation of these map representations to support the integration with active perception. Our approaches enable robots to autonomously explore unknown environments and gather informative measurements to incrementally build high-fidelity maps.

We begin by investigating GPs in active perception for robot mapping. In particular, we propose a novel GP-based scalar field mapping approach that utilizes GPs to model the spatial distribution of environmental properties, as detailed in Chapter 3. The goal of this work is to minimize the uncertainty in regions of interest in an unknown environment, thereby achieving high mapping accuracy in these regions. Given that GPs inherently provide uncertainty modeling suitable for active perception, we address our research question from the perspective of computational efficiency. To facilitate online incremental mapping, we initialize a spatially correlated grid map with the GP prior from our model, and perform sequential Bayesian fusion to incorporate new measurements over time. We leverage the uncertainty modeling capability of GPs to formulate an active perception strategy, where the robot selects viewpoints to maximize expected uncertainty reduction in regions of interest. This is achieved by forward-simulating map updates and evaluating the resulting posterior distributions. A key contribution is the introduction of an integral kernel for the underlying GP model, enabling the maintenance of an adaptive resolution map in both a computationally efficient and theoretically sound manner. Our proposed integral kernel formulation enables an adaptive strategy to preserve the probabilistic nature of GPs while reducing the grid map resolution where high details are not required. This leads to significantly lower memory usage and, more importantly, faster inference for forward simulation, which is crucial for efficient view planning. We demonstrate the effectiveness of this approach in a 2D temperature field mapping application using an unmanned aerial vehicle (UAV) equipped with a thermal sensor.

The second contribution is an active perception approach based on uncertainty estimation in image-based neural rendering, referred to as NeU-NBV in Chapter 4. Image-based neural rendering employs a pretrained network that, given a set of posed RGB reference images, synthesizes photorealistic views from novel viewpoints. Our goal is to actively collect RGB measurements as references in an unknown scene to enhance the rendering performance of the neural network. The key innovation of this work lies in modeling the color rendering process probabilistically, allowing us to acquire rendering uncertainty based on the predicted

variance of color rendering at each pixel, conditioned on the current reference images. The uncertainty exposes the areas in the scene where the network is less confident in its rendering using the current reference image collection, and can thus be used to inform the view planning process. We leverage the uncertainty estimation to formulate an NBV planning problem, directing the robot to sequentially acquire new measurements at viewpoints with the highest predicted uncertainty. These accumulated image measurements, together with the rendering network, serve as the internal map representation, enabling our approach to retrieve scene information from novel viewpoints. Therefore, our approach allows robots to actively gather informative measurements to more accurately represent the scene, without the need for explicitly updating a map representation online.

The third contribution is a semantic-targeted active perception approach for robot mapping, called STAIR and presented in Chapter 5. In this chapter, we propose a novel active perception strategy for selectively reconstructing specific object classes in an unknown environment, while deprioritizing semantically irrelevant regions of the scene. This is particularly important in scenarios where a robot's attention should be focused on task-relevant objects to make the best use of limited mission resources. To achieve this, we integrate a NeRF with semantic information as our map representation, enabling dense semantic modeling of the scene. We derive uncertainty estimates based on the density distribution in NeRFs, which allows us to identify the areas with high geometric ambiguity. With the help of dense semantic and uncertainty rendering at novel viewpoints, our view planning method can actively select informative viewpoints for collecting new measurements, improving the reconstruction quality of target objects. This targeted strategy significantly enhances mapping efficiency in semantic-targeted tasks by focusing measurement acquisition on relevant parts of the scene, without wasting resources on semantically irrelevant regions.

The fourth contribution is an active scene-level reconstruction approach based on GS, which we refer to as ActiveGS in Chapter 6. In contrast to the NeRF-based representations utilized in Chapter 4 and Chapter 5, which require computationally heavy dense sampling for volume rendering to synthesize views, this chapter addresses scene-level photorealistic mapping by leveraging more efficient GS as the core map representation. Since GS primarily models scene surfaces and lacks holistic spatial information, we propose a hybrid map representation that combines a GS map with a coarse voxel map, leveraging the strengths of both representations: the high-fidelity scene reconstruction capabilities of GS and the spatial modeling strengths of the voxel map. At the core of our approach is an effective confidence modeling technique that assigns confidence values to each Gaussian primitive in the GS map based on the viewpoint distribution. This allows our approach to identify under-reconstructed areas in the GS map for fur-

ther inspection. In parallel, we utilize spatial information from the voxel map to target unexplored areas and assist in collision-free path planning. Our approach actively collects measurements in both under-reconstructed and unexplored areas for map updates, achieving superior GS reconstruction results in indoor scenarios.

Together, these four contributions form the foundation of this thesis, each offering a distinct approach to active perception for robot mapping. By leveraging different learning-based map representations, including GPs, image-based neural rendering networks, semantic NeRFs, and GS, we investigate utility formulations tailored to these representations and associated mapping objectives. Our proposed methods enable efficient view planning in active perception settings, and we evaluate their performance in both simulation and real-world scenarios. The results demonstrate their effectiveness in enhancing mapping efficiency and quality compared to baseline approaches, marking a significant advancement in the field of active perception for learning-based robot mapping.

1.3 Publications

Parts of this thesis have been published in the following peer-reviewed conference papers and journal articles:

- Liren Jin, Julius Rückin, Stefan 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.

- 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, 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. DOI: 10.1109/IROS58592.2024.10801401.

- Liren Jin, Xingguang Zhong, Yue Pan, Jens Behley, Cyrill Stachniss, and Marija Popović. ActiveGS: Active Scene Reconstruction Using Gaussian Splatting. *IEEE Robotics and Automation Letters (RA-L)*, 10(5):4866-4873, 2025. DOI: 10.1109/LRA.2025.3555149.

1.4 Collaborations

During my doctoral studies, I also contributed as a co-author to the following publications, which are not included in this thesis:

- 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 and Automation (ICRA)*, 2022. DOI: 10.1109/ICRA46639.2022.9812025.

- 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.

- Sicong Pan*, Liren Jin*, Hao Hu, Marija Popović, and Maren Bennewitz. How Many Views Are Needed to Reconstruct an Unknown Object Using NeRF? In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2024. DOI: 10.1109/ICRA57147.2024.10610617. (* authors contributed equally).

- Hao Hu, Sicong Pan, Liren Jin, Marija Popović, and Maren Bennewitz. Active Implicit Reconstruction Using One-Shot View Planning. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2024. DOI: 10.1109/ICRA57147.2024.10611542.

- Sicong Pan*, Liren Jin*, Xuying Huang, Cyrill Stachniss, Marija Popović, and Maren Bennewitz. Exploiting Priors from 3D Diffusion Models for RGB-Based One-Shot View Planning. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2024. DOI: 10.1109/IROS58592.2024.10802551. (* authors contributed equally).

- Yue Pan, Xingguang Zhong, Liren Jin, Louis Wiesmann, Marija Popović, Jens Behley, and Cyrill Stachniss. PINGS: Gaussian Splatting Meets Distance Fields within a Point-Based Implicit Neural Map. In *Proc. of Robotics: Science and Systems (RSS)*, 2025. DOI: 10.15607/RSS.2025.XXI.040.

- Sicong Pan, Liren Jin, Xuying Huang, Cyrill Stachniss, Marija Popović, and Maren Bennewitz. DM-OSVP++: One-Shot View Planning Using 3D Diffusion Models for Active RGB-Based Object Reconstruction. *arXiv preprint*, 2025. DOI: 10.48550/arXiv:2504.11674.

- Xingguang Zhong, Yue Pan, Liren Jin, Marija Popović, Jens Behley, and Cyrill Stachniss. Globally Consistent RGB-D SLAM with 2D Gaussian Splatting. *arXiv preprint*, 2025. DOI: 10.48550/arXiv:2506.00970.
- Xingguang Zhong, Liren Jin, Marija Popović, Jens Behley, and Cyrill Stachniss. Dynamic Visual SLAM Using a General 3D Prior. *arXiv preprint*, 2025. DOI: 10.48550/arXiv:2512.06868.

1.5 Open Source Contributions

To facilitate further research in active perception for learning-based robot mapping, we have open-sourced the implementations of all proposed approaches presented in this thesis:

- **ARGPF-Mapping**: Adaptive-Resolution Field Mapping Using Gaussian Process Fusion with Integral Kernels
  `https://github.com/dmar-bonn/argpf-mapping`
- **NeU-NBV**: Next Best View Planning Using Uncertainty Estimation in Image-Based Neural Rendering
  `https://github.com/dmar-bonn/neu-nbv`
- **STAIR**: Semantic-Targeted Active Implicit Reconstruction
  `https://github.com/dmar-bonn/stair`
- **ActiveGS**: Active Scene Reconstruction Using Gaussian Splatting
  `https://github.com/dmar-bonn/active-gs`

Chapter 2

Basic Techniques

In this chapter, we review basic techniques essential for understanding the research question and the main contributions of this thesis. We aim to provide a self-contained reference for the reader, while noting that we do not claim any technical contributions for the techniques presented in this chapter. The key concepts throughout this thesis are learning-based map representations and active perception strategies for robot mapping. We start by introducing common map representations used in robotic applications, comparing their respective strengths and limitations. We then shift our focus to recent developments in learning-based mapping techniques, with a detailed overview of the learning-based map representations adopted in this thesis. Following that, we discuss fundamental approaches and methodologies for actively building map representations using robots, with a particular focus on how different mapping objectives and characteristics of the chosen map representations influence the design of active perception approaches.

2.1 Map Representations

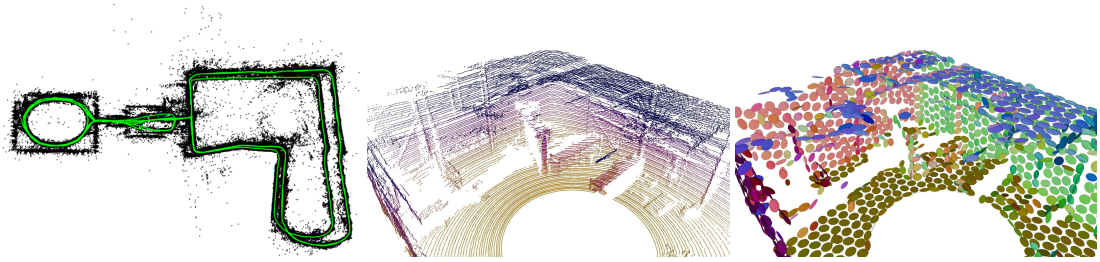
A map representation is a structured abstraction of the surrounding environment that captures geometric information and spatial relationships between objects and, in many cases, also textural or semantic information. From simple 2D layouts such as floor plans to dense 3D reconstructions and semantically rich models, map representations vary in their complexity and the type of information they encode, reflecting the needs of the application.

Even for us humans, map representations are an integral part of our daily lives. Tourists rely on city maps displayed on smartphones to effortlessly navigate busy streets and locate their goal destinations. Game players interact with vivid 3D models in video games and virtual environments, acquire enhanced entertainment

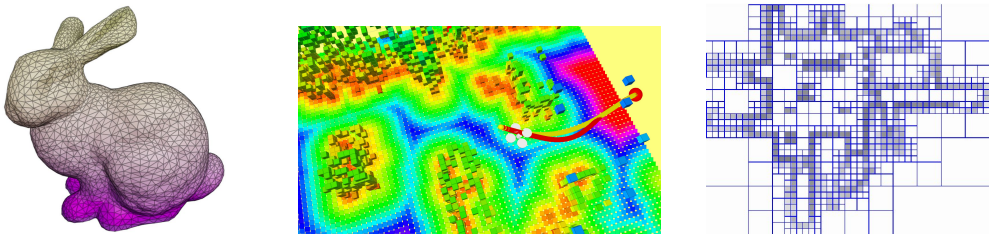
through immersive virtual experiences. Engineers use terrain maps in geographic information systems to analyze potential land use, plan urban development, and monitor environmental changes. Farmers leverage high-resolution modeling of crop fields to optimize their operations in precision agriculture. Each type of these map representations, although it appears in different formats, serves to simplify and communicate spatial information for human decision-making.

More closely aligned with the research focus of this thesis, we concentrate on map representations used for robot mapping. At the core of robot mapping lies the processing of onboard sensor measurements, such as from LiDAR, RGB, or depth cameras, to construct a spatial representation of the environment. These representations underpin a robot's spatial awareness, crucial for enabling robots to understand and interact effectively with their surroundings. Such spatial awareness is fundamental for a wide range of robotic tasks such as state estimation, scene understanding, planning, and physical interaction [103]. In addition to enabling autonomous robot behavior, these map representations also offer valuable insights to human operators, assisting in decision-making for tasks such as industrial inspection, search and rescue, and urban monitoring. Depending on the specific task requirements, operational conditions, and available sensors, various types of map representations can be employed.

The most commonly used type of map representation in robotics is the metric map, focusing on preserving scene geometry or texture, as shown in Figure 2.1. Geometric feature points are sparse, distinctive points that can be reliably detected and matched across different viewpoints. Their robustness makes them popular for localization tasks [98,111]. Point clouds, with their simple data structure, are widely used to capture scene surfaces using unstructured 3D points from LiDAR or depth cameras, or reconstructed from RGB inputs using multi-view stereo algorithms. They are largely applied in tasks like localization [34, 206], object detection [212,220], and scene understanding [190,197,209], but can struggle to preserve fine details when sampling is sparse. Surfel maps extend point clouds by using 2D disks to model local surface patches with position and orientation, offering denser models [171, 193, 198]. Surface meshes, composed of connected polygons, typically triangles, provide explicit connectivity for better surface approximations. While they are preferred for high-quality geometric and textural modeling [6, 131, 179], they also introduce complexity in connectivity maintenance, which complicates incremental mapping. Volumetric maps use voxel grids to maintain spatial data like occupancy or signed distance to surfaces [3, 73, 119, 125]. They can handle both surface and free space information, crucial for planning and navigation. However, they are often constrained by their fixed resolution, leading to either excessive memory usage for high-resolution maps or loss of details for low-resolution maps. To address this, hierarchical



(a) Geometric features [111] (b) Point cloud [137] (c) Surfels [137]



(d) Surface mesh [50] (e) Voxel grid [48] (f) Multi-resolution grid [82]

Figure 2.1: Examples of conventional map representations. (a) Sparse geometric feature points commonly used for localization purposes. (b) A point cloud map provides sampling on the scene surfaces. (c) Surfels extend points to 2D disks, capturing both position and orientation of local surface patches. (d) Surface meshes approximate surfaces using connected polygons. (e) Volumetric maps utilize voxel grids to maintain spatial information such as occupancy or signed distance to the surfaces. (f) Hierarchical volumetric maps adaptively allocate resolution to different regions of the scene based on their complexity. These conventional map representations rely on rigid, prefixed shape primitives, limiting the map granularity for representing details in the scene.

approaches like octrees [22, 44, 59, 150, 184] or VDB-based maps [7, 112, 188] dynamically adjust resolution based on scene complexity, optimizing memory and efficiency. Voxel hashing [33, 115], on the other hand, reduces mapping cost by only spawning voxels as needed. Despite their versatility and effectiveness, conventional metric map representations relying on rigid, predefined shape primitives remain limited in reconstruction granularity and face a trade-off between map fidelity and memory efficiency.

In many robotic applications, high-level semantic reasoning is crucial for tasks like planning and interacting with complex environments, which cannot be fully achieved with metric maps alone. For instance, robot manipulation requires recognizing specific objects for tasks such as grasping or assembly, while context-aware navigation requires interpreting the semantic meaning of regions to plan paths. In these scenarios, maps must also encode semantic information, as illustrated in Figure 2.2. To achieve this, metric-semantic maps augment the above-mentioned metric maps with semantic information [14, 105, 174, 176, 217]. Seman-

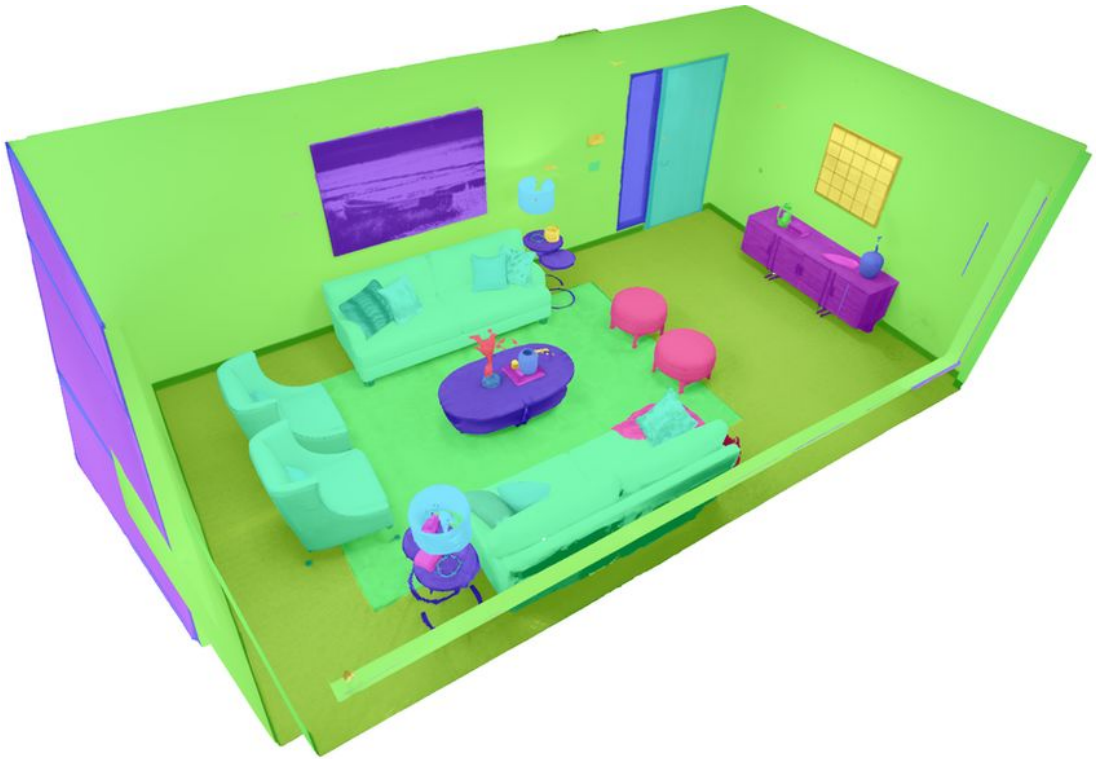


Figure 2.2: An example of semantics in map representations. Different colors represent different semantic categories, such as walls, floors, sofas, and doors. This semantic information is crucial for high-level reasoning in robotic applications, enabling robots to understand and interact with their environments more effectively. Semantic maps can be constructed by fusing semantic segmentation results from images or by training neural networks to predict semantic labels directly from the 3D map itself. Image from Straub *et al.* [169].

tic information can be integrated by fusing image-level semantic segmentation from pretrained networks or training networks to directly predict semantic labels from the 3D map, such as in point cloud semantic segmentation [37, 84, 136]. The segmentation networks are usually trained offline on large, semantically annotated datasets, and then deployed for online labeling, forming the foundation of metric-semantic mapping [23, 108, 170].

Besides metric and metric-semantic maps, several specialized map representations have been developed to address the unique requirements in robotic applications. For example, topological maps are well-suited for high-level navigation tasks [13, 39], where the focus lies in connectivity and spatial relationships between objects rather than geometric accuracy. Dynamic scene representations are essential for environments involving moving objects [9, 135, 153], enabling tracking and prediction over time. Additionally, map representations for deformable or articulated objects [41, 42] provide structure-aware representations that are critical for applications such as manipulation or human-robot interaction.

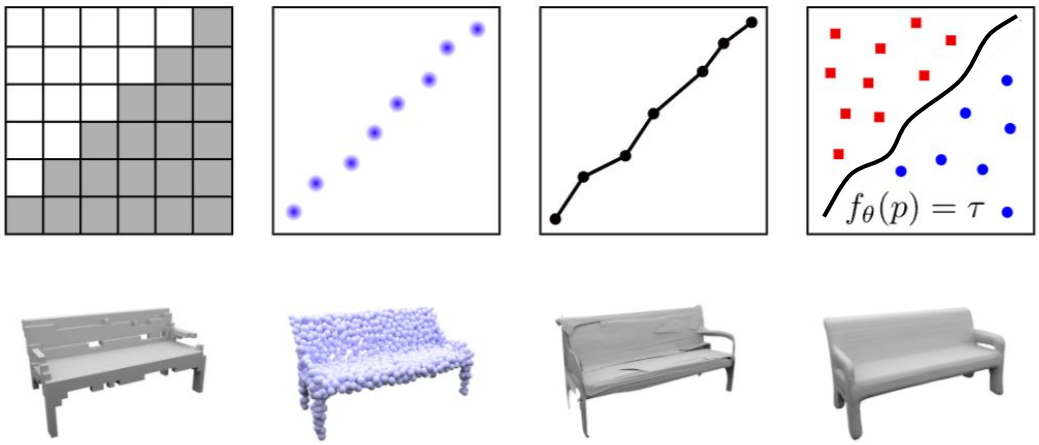


Figure 2.3: A simple example of comparison between conventional and learning-based map representations. Conventional dense map representations utilize (a) voxels, (b) points, or (c) meshes to represent the scene. The granularity of the reconstructed scene is largely limited by its spatial discretization. Learning-based dense mapping techniques utilize data-driven approaches, such as neural networks, to learn the scene representation directly from measurement data, enabling continuous, resolution-independent modeling of complex scenes. Image adapted from Mescheder *et al.* [106].

In this thesis, our goal is to accurately model static, unknown environments. We utilize learning-based map representations that can represent the scene in a continuous manner, allowing for inference at arbitrary resolution and preservation of fine-grained environmental details. Instead of being explicitly constructed using rigid primitives such as point clouds, voxel grids, or surface meshes, these learning-based map representations allow for direct optimization of the map to account for sensor measurements. By leveraging data-driven approaches, e.g., neural networks or optimizable shape primitives, they encode information of the scene properties in a continuous, resolution-independent manner, leading to more flexible and expressive modeling of complex scenes. We show a toy example comparing conventional and learning-based map representations in Figure 2.3, highlighting the key difference in how spatial information is preserved. In the following subsections, we introduce the basic techniques of the learning-based map representations used in this thesis.

2.1.1 Gaussian Processes

GP models are among the earliest and most influential learning-based map representations. In geostatistics, they are known as kriging [120], a method for spatial mapping of geologic properties. More generally, GPs represent the mapping tar-

get, such as temperature, gas density, or signed distance, as a distribution in the function space. In other words, a GP can be interpreted as a collection of random variables, any finite subset of which follows a joint Gaussian distribution [140].

A GP model is fully described by its mean function $\mu(\mathbf{x})$ indicating the expected function value at an input point $\mathbf{x}$, and kernel function $k(\mathbf{x}, \mathbf{x}')$ describing the spatial correlations between the function values at $\mathbf{x}$ and $\mathbf{x}'$. We denote a GP as $f(\mathbf{x}) \sim \mathcal{GP}(\mu(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$ with:

$$\mu(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})], \tag{2.1}$$

$$k(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(f(\mathbf{x}) - \mu(\mathbf{x}))(f(\mathbf{x}') - \mu(\mathbf{x}'))]. \tag{2.2}$$

The kernel function is central to a GP, and different kernel formulations can be selected based on the attributes of the mapping target, such as commonly used squared exponential or Matérn kernels [140]. Although the kernel function is manually selected, its hyperparameters are typically learned from sensor measurements collected during online mapping missions or from previously acquired data in a similar domain. Given a set of measurements collected at locations $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$ with corresponding noisy measurement values $\mathbf{y} = \{y_i\}_{i=1}^N$, where $y_i = f(\mathbf{x}_i) + \epsilon$ and $\epsilon \sim \mathcal{N}(0, \sigma^2)$, the hyperparameters can be determined via maximizing the marginal log likelihood:

$$\mathbf{w}' = \arg\max_{\mathbf{w}} \ \log p(\mathbf{y} \mid \mathbf{X}, \mathbf{w}), \tag{2.3}$$

with:

$$\log p(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = -\tfrac{1}{2}\mathbf{y}^\top (\mathbf{K}_{\mathbf{X},\mathbf{X}} + \sigma^2 \mathbf{I})^{-1}\mathbf{y} - \tfrac{1}{2}\log|\mathbf{K}_{\mathbf{X},\mathbf{X}} + \sigma^2\mathbf{I}| - \tfrac{N}{2}\log(2\pi), \tag{2.4}$$

where $\mathbf{K}_{\mathbf{X},\mathbf{X}} = k(\mathbf{X}, \mathbf{X}; \mathbf{w})$ is the covariance matrix over the measurement locations, parameterized by the hyperparameters $\mathbf{w}$.

Given the learned kernel function, we can infer the function distribution at any query points via GP regression [140]. The joint distribution over measurements values $\mathbf{y}$ and the function value $\mathbf{f}_* = f(\mathbf{X}_*)$ at a set of query point $\mathbf{X}_* = \{\mathbf{x}_j\}_{j=1}^M$ is a joint Gaussian distribution:

$$\begin{bmatrix} \mathbf{y} \\ \mathbf{f}_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_{\mathbf{X}} \\ \boldsymbol{\mu}_{\mathbf{X}_*} \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{\mathbf{X},\mathbf{X}} + \sigma^2\mathbf{I} & \mathbf{K}_{\mathbf{X},\mathbf{X}_*} \\ \mathbf{K}_{\mathbf{X}_*,\mathbf{X}} & \mathbf{K}_{\mathbf{X}_*,\mathbf{X}_*} \end{bmatrix} \right), \tag{2.5}$$

where $\boldsymbol{\mu}_{\mathbf{X}} = \mu(\mathbf{X})$ and $\boldsymbol{\mu}_{\mathbf{X}_*} = \mu(\mathbf{X}_*)$ represent the mean value vectors; $\mathbf{K}_{\mathbf{X},\mathbf{X}} = k(\mathbf{X}, \mathbf{X}; \mathbf{w}')$, $\mathbf{K}_{\mathbf{X},\mathbf{X}_*} = k(\mathbf{X}, \mathbf{X}_*; \mathbf{w}')$, and $\mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) = k(\mathbf{X}_*, \mathbf{X}_*; \mathbf{w}')$ denote the covariance matrices over the measurement locations, between the measurement locations and the query points, and over the query points, respectively.

The predictive distribution of $\mathbf{f}_*$ at $\mathbf{X}_*$ conditioned on $\mathbf{X}$ and $\mathbf{y}$ is then:

$$p(\mathbf{f}_* \mid \mathbf{X}, \mathbf{y}, \mathbf{X}_*) = \mathcal{N}\big(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*\big), \tag{2.6}$$

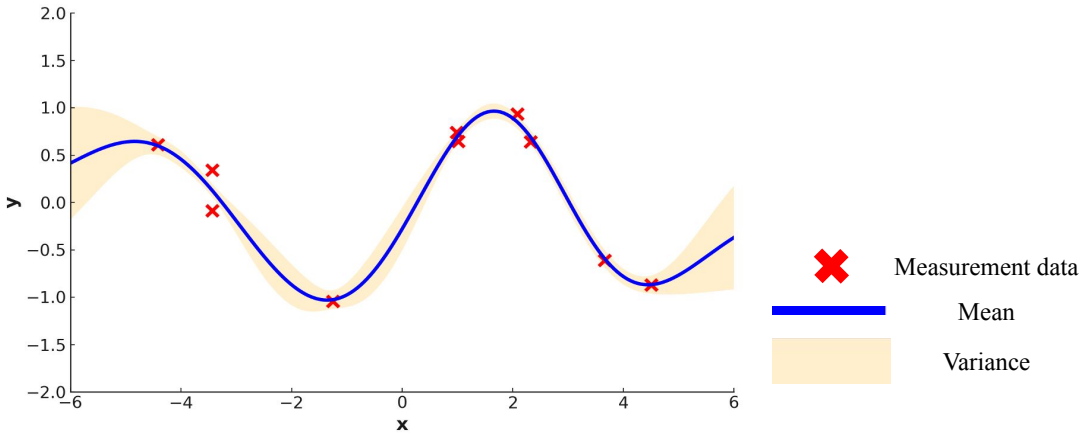


Figure 2.4: A 1D example of GPs. The x-axis represents the location of measurements, and the y-axis indicates their values. Given point measurements shown as red crosses, GPs enable inference at any locations, yielding both mean and variance predictions. Areas with sparse measurements often show higher variance, indicating high map uncertainty. Active perception approaches based on GPs can utilize this attribute to formulate utility evaluation to guide view planning toward uncertain areas.

with:

$$\boldsymbol{\mu}_* = \boldsymbol{\mu}_{\mathbf{X}_*} + \mathbf{K}_{\mathbf{X}_*,\mathbf{X}}\left[\mathbf{K}_{\mathbf{X},\mathbf{X}} + \sigma^2\mathbf{I}\right]^{-1}(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{X}}), \tag{2.7}$$

$$\boldsymbol{\Sigma}_* = \mathbf{K}_{\mathbf{X}_*,\mathbf{X}_*} - \mathbf{K}_{\mathbf{X}_*,\mathbf{X}}\left[\mathbf{K}_{\mathbf{X},\mathbf{X}} + \sigma^2\mathbf{I}\right]^{-1}\mathbf{K}_{\mathbf{X},\mathbf{X}_*}. \tag{2.8}$$

We illustrate a 1D example of GPs in Figure 2.4, showing the posterior distribution of the underlying function given a set of noisy measurements. GPs offer a powerful framework for continuous and probabilistic mapping, and have therefore been successfully applied to various robot mapping tasks, such as occupancy [65, 79], terrain [25, 183], pipe thickness [185], gas distribution [167], and signed distance [199]. The inherent uncertainty formulation in the form of the variance prediction as given by Equation (2.8) can be directly utilized for active perception. Several works adopt the uncertainty information in GP-based mapping for exploration [66, 178]. However, GPs can be computationally expensive, especially for mapping in higher dimensions with dense measurements, due to the need to invert covariance matrices in Equation (2.7) and Equation (2.8) for inferring spatial correlations. This often limits GPs to relatively small-scale environments or requires approximation techniques to handle the computational complexity [100, 159, 168, 183].

We introduce a novel online incremental mapping approach based on GPs in Chapter 3, where we improve its mapping efficiency while maintaining the probabilistic inference capability important for active perception.

2.1.2 Neural Radiance Fields

A growing body of research has focused on learning-based dense map representations specifically designed for complex 3D reconstruction. Pioneering works in this domain leverage neural networks to predict geometric attributes such as occupancy probabilities [106, 130], or signed distance values [121, 128], at arbitrary query points, therefore enabling continuous representation of the scene geometry. These implicit representations are either optimized directly for a specific scene or trained offline using large datasets of 3D shapes, allowing them to generalize well to unseen scenes during inference.

Building on the idea of using neural networks to represent scenes, NeRFs [107] have emerged as a powerful approach for scene modeling. They encode the entire scene into the network's weights, enabling the preservation of both accurate geometry and photorealistic textures. Combined with volume rendering techniques, NeRFs enable novel view synthesis with high visual realism while maintaining a compact memory footprint. The core of NeRF's success lies in the differentiability of the rendering process, which allows the network to be trained by minimizing the discrepancy between rendered views and the ground-truth images observed from densely sampled viewpoints in a scene.

A NeRF can be parameterized by a neural network $f : (\mathbf{x}, \mathbf{d}) \rightarrow (\mathbf{c}, \sigma)$ implemented as a multi-layer perceptron (MLP). This MLP maps a 3D position $\mathbf{x} = (x, y, z) \in \mathbb{R}^3$ together with a 2D viewing direction (θ, ϕ), represented by a unit vector $\mathbf{d} \in \mathbb{S}^3$, to an RGB color $\mathbf{c}(\mathbf{x}, \mathbf{d}) \in \mathbb{R}^3$ and a volume density value $\sigma(\mathbf{x}) \in \mathbb{R}_{\geq 0}$. Note that, different from the previous section, where σ denotes the variance in GP models, the σ used here does not carry a probabilistic meaning. For a camera ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ passing through an image plane, where $\mathbf{o} \in \mathbb{R}^3$ is the camera origin and t is the distance along the ray, the corresponding pixel color $\mathbf{C}(\mathbf{r})$ can be computed using the volume rendering equation [74]:

$$\mathbf{C}(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\, \sigma(\mathbf{r}(t))\, \mathbf{c}(\mathbf{r}(t), \mathbf{d})\, dt, \tag{2.9}$$

where t_n and t_f denote near and far bounds of the ray, and $T(t)$ is the accumulated transmittance defined as:

$$T(t) = \exp\left(- \int_{t_n}^{t} \sigma(\mathbf{r}(s))\, ds \right), \tag{2.10}$$

representing the probability that the ray travels from t_n to t without termination. In practice, this integral is approximated via stratified dense sampling along each ray, where $[t_n, t_f]$ is partitioned into N evenly-spaced bins, and one sample is drawn uniformly at random from each bin for each training iteration as:

$$t_i \sim \mathcal{U}\big[t_n + \tfrac{i-1}{N}(t_f - t_n),\ t_n + \tfrac{i}{N}(t_f - t_n)\big]\,. \tag{2.11}$$

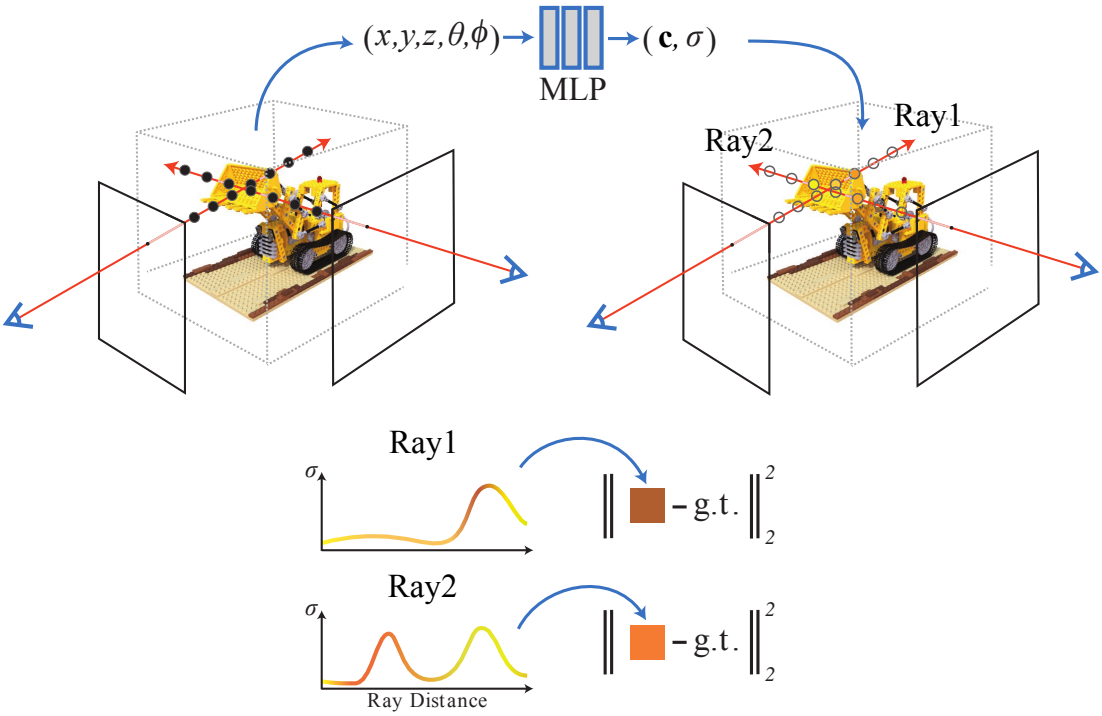


Figure 2.5: A NeRF preserves global scene information in an MLP. Given a point with a viewing direction, this MLP predicts color and density value. To render a novel view, NeRFs densely sample points along the camera ray and query the MLP for each sampling point. These per-point colors and densities compose the final pixel color via differentiable volume rendering. The differentiability allows the MLP to be optimized given ground-truth RGB color measurements. Image from Mildenhall *et al.* [107]

This stratified sampling strategy allows NeRFs to learn a continuous scene representation, since it results in the MLP being evaluated at continuous positions during training. Consequently, the volume rendering can be reformulated as:

$$\mathbf{C}(\mathbf{r}) \approx \sum_{i=1}^{N} T_i \left(1 - \exp(-\sigma(\mathbf{r}(t_i))\,\delta_i)\right) \mathbf{c}(\mathbf{r}(t_i), \mathbf{d}), \tag{2.12}$$

where $\delta_i = t_{i+1} - t_i$ is the spacing between consecutive samples. The accumulated transmittance is recursively defined as:

$$T_i = \prod_{j=1}^{i-1} \exp(-\sigma(\mathbf{r}(t_j))\,\delta_j)\,. \tag{2.13}$$

Since the volume rendering process is differentiable, we can supervise the NeRFs training with the loss:

$$\mathcal{L} = \frac{1}{|\mathcal{R}|} \sum_{\mathbf{r}\in\mathcal{R}} \left\|\mathbf{C}(\mathbf{r}) - \hat{\mathbf{C}}(\mathbf{r})\right\|_2^2, \tag{2.14}$$

where $\mathcal{R}$ is the set of rays across all training views, and $\hat{\mathbf{C}}(\mathbf{r})$ represents the ground-truth color corresponding to each ray. By updating the MLP's weights

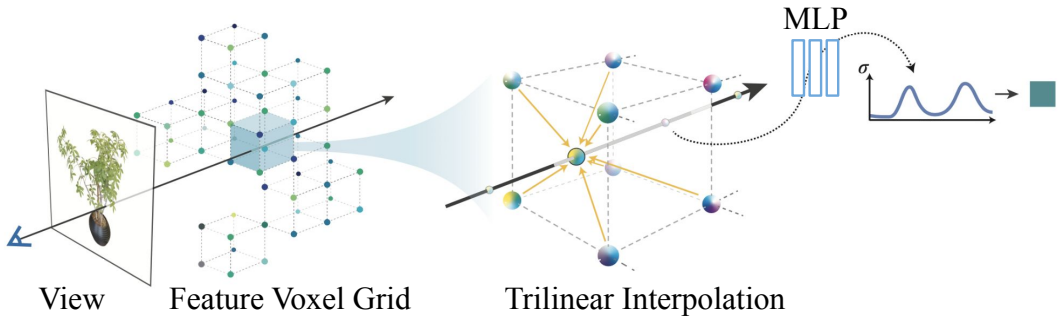


Figure 2.6: We illustrate a hybrid NeRF representation. For rendering a novel view, the features of dense sampling points can be retrieved from the voxel grid via trilinear interpolation. An MLP then interprets these features into density and color information for volume rendering. Since the scene information is preserved locally in the feature voxel grid, hybrid NeRFs mitigate the forgetting issues commonly seen in vanilla NeRFs during incremental mapping. Image adapted from Fridovich-Keil *et al.* [43].

to account for training views, NeRFs achieve highly detailed reconstructions with accurate geometry and photorealistic textures. We show an illustration of the NeRFs pipeline in Figure 2.5.

While vanilla NeRFs offer an excellent trade-off between reconstruction quality and memory efficiency, their representational capacity is inherently constrained by the network size. This limitation often leads to over-smoothing artifacts in large-scale scenes, where a single global network struggles to preserve fine details across extended spatial extents. Moreover, NeRFs are not naturally suited for incremental updates, since the network tends to forget previously learned information when updated with new measurements, a phenomenon known as catastrophic forgetting [172]. This poses a major challenge for active perception for robot mapping, which often necessitates online map updates. To address these limitations, recent research has proposed hybrid NeRF representations that combine shallow neural networks with spatially structured feature voxel grids [110, 173], as shown in Figure 2.6. These methods store local scene information in the form of optimizable feature vectors in voxel grids. A lightweight MLP is then employed to interpret features at arbitrary sampling points, which are retrieved through trilinear interpolation in the voxel grid, into color and volume density of the radiance field. This design improves scalability and allows for localized updates, making them more suitable for online incremental mapping scenarios. We utilize a hybrid NeRF representation and integrate semantic information to achieve semantic-targeted active reconstruction in Chapter 5.

In parallel, another line of research aims to avoid the time-consuming per-scene optimization of NeRFs by leveraging image-based neural rendering techniques [194, 207]. Unlike classical rendering techniques, which project explicit

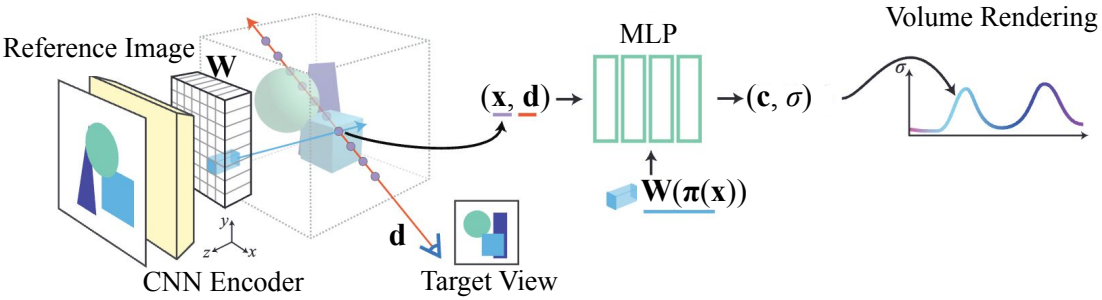


Figure 2.7: We show how image-based neural rendering works. A pretrained image encoder first encodes the reference images to acquire their feature maps. To render a target view, image-based neural rendering methods sample dense points along the ray, which are projected to each reference image plane via a projection operation $\boldsymbol{\pi}$ to acquire corresponding image features via bilinear interpolation. Features collected from different reference images are fused, e.g., by averaging. The final features are then decoded by an MLP into color and density, which compose the pixel color via volume rendering. By training on large datasets, image-based neural rendering learns to render target views by conditioning on reference images, allowing for generalizable novel view synthesis. Image adapted from Yu *et al.* [207].

3D content, e.g., surface meshes, onto 2D image planes, image-based rendering directly generates novel views by warping and compositing an existing set of reference images. Image-based neural rendering extends this idea by employing neural networks that condition novel view synthesis on features extracted from nearby reference images [180]. Specifically, reference images are first encoded into feature maps, from which features corresponding to each query point are retrieved via projection and bilinear interpolation. The features collected across reference images are then fused, e.g., by averaging, to form the final feature vector of the query point, which is subsequently decoded by an MLP into color and density. The volume rendering still follows Equation (2.12) to acquire final rendering results. Trained on large datasets of multi-view images, these models learn strong priors that enable generalization to new scenes and support direct novel view synthesis from only a few reference images at inference time. We illustrate the image-based neural rendering process in Figure 2.7. However, similar to NeRFs, these methods also require dense sampling in the scene for volume rendering, which is computationally expensive. In Chapter 4, we mitigate the inefficiency in dense sampling and propose incorporating uncertainty modeling in image-based neural rendering to guide view planning for active perception.

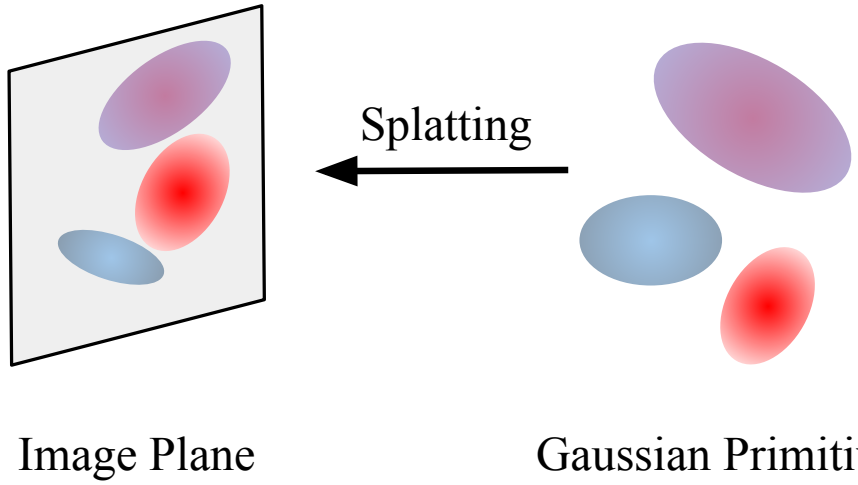


Figure 2.8: GS maps represent the scene using a set of explicit but optimizable shape primitives. Different from NeRFs that require dense sampling along rays to estimate the color and density, the primitives in GS maps can be directly projected onto image planes for view synthesis, enabling more efficient rendering compared to NeRF-based approaches. In contrast to conventional map representations that rely on rigid shape primitives, GS maps integrate differentiability into the rendering pipeline, allowing them to be optimized directly from scene measurements.

2.1.3 Gaussian Splatting

Recently, GS maps have appeared as a promising alternative to NeRF-based approaches for photorealistic reconstruction. Different from NeRFs that rely on dense sampling to acquire scene information in the space, GS introduces explicit radiance fields constructed from a set of optimizable shape primitives [29, 61, 78], referred to as Gaussian primitives. Each primitive is defined by its parameters $\mathbf{g}_i = (\mathbf{x}_i\,, \mathbf{q}_i\,, \mathbf{s}_i,\, \mathbf{c}_i\,, o_i)$, where $\mathbf{x}_i \in \mathbb{R}^3$ denotes the position of the primitive center; $\mathbf{q}_i \in \mathbb{R}^4$ is its rotation in the form of a quaternion; $\mathbf{s}_i = [s_i^x\,, s_i^y\,, s_i^z] \in \mathbb{R}^3_+$ represents the scaling factors along the three axes of the primitive; $\mathbf{c}_i(\mathbf{d}) \in [0\,, 1]^3$ represents the RGB color, decoded from the viewing direction, e.g., via spherical harmonics [43]; $o_i \in [0\,, 1]$ is the opacity value. Intuitively, a Gaussian primitive can be viewed as an ellipsoid whose opacity decreases with distance from its center, as illustrated in Figure 2.8. Its distribution in the world coordinate is formulated as:

$$\mathcal{N}(\mathbf{x}; \mathbf{x}_i, \mathbf{\Sigma}_i) = \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{x}_i)^\top \mathbf{\Sigma}_i^{-1}(\mathbf{x}-\mathbf{x}_i)\right)\,, \tag{2.15}$$

where $\mathbf{\Sigma}_i = \mathbf{R}(\mathbf{q}_i)\,\mathrm{diag}\left((s_i^x)^2, (s_i^y)^2, (s_i^z)^2\right)\,\mathbf{R}(\mathbf{q}_i)^\top$ is the covariance matrix representing the shape of the Gaussian primitive in 3D space, with the rotations matrix $\mathbf{R}(\mathbf{q}_i) \in SO(3)$ derived from the quaternion $\mathbf{q}_i$.

For rendering, each primitive is first projected onto a 2D image plane via the elliptical weighted average filter [223]. The projected covariance matrix can be formulated as:

$$\mathbf{\Sigma}_i' = \mathbf{J}_i \mathbf{W}_i \, \mathbf{\Sigma}_i \, \mathbf{W}_i^\top \mathbf{J}_i^\top, \tag{2.16}$$

where $\mathbf{J}_i$ is the Jacobian of the affine approximation of the projective transformation and $\mathbf{W}_i$ is the viewing transformation from the world coordinate to the image coordinate. Consequently, the opacity of a projected Gaussian primitive $\mathbf{g}_i$ at a pixel $\mathbf{u}$ on the image plane can be expressed as:

$$\alpha_i(\mathbf{u}) = \exp\left(-\frac{1}{2}(\mathbf{u}-\mathbf{u}_i)^\top \mathbf{\Sigma}_i'^{-1}(\mathbf{u}-\mathbf{u}_i)\right) \, o_i, \tag{2.17}$$

where $\mathbf{u}_i$ is the projected position of the primitive's center on the image plane. These projected Gaussian primitives are ordered based on their distance to the image plane, and the final pixel color can be acquired by blending all Gaussian primitives overlapping the pixel from near to far, a technique commonly referred to as differentiable rasterization:

$$\mathbf{C}(\mathbf{u}) = \sum_{i=1}^{n} w_i \mathbf{c}_i \, , w_i = T_i \alpha_i \, , T_i = \prod_{j<i} (1-\alpha_j). \tag{2.18}$$

By supervising the rendering results with training views, similar to Equation (2.14), we can optimize the parameters of Gaussian primitives.

GS maps preserve the scene information explicitly in the primitives and do not require dense sampling to query scene information, leading to faster novel view synthesis. While compared to conventional explicit map representations, which typically lack end-to-end optimization capabilities, GS retains differentiability throughout the rendering pipeline. This allows the geometric and textural parameters of GS maps to be optimized to account for measurements, enabling high-fidelity reconstructions without the computational overhead associated with the dense sampling in NeRFs. As a result, GS can produce photorealistic renderings at interactive frame rates, making it particularly suitable for online robotic applications. Beyond rendering efficiency, the explicit nature of the representation facilitates direct manipulation, fusion, and incremental updates, capabilities that are important for dynamic or large-scale mapping tasks [21]. We adopt GS as the main map representation for active scene-level reconstruction in Chapter 6.

2.2 Active Perception for Robot Mapping

Active perception in robotics refers to the process of actively planning and moving sensor viewpoints to acquire informative measurements for a given task. Unlike

passive perception, which relies on fixed, predefined perception strategies or manual control, active perception automatically adapts the perception process based on current knowledge and task requirements. This is an important capability when performing robotic tasks, e.g., localization, object detection, semantic scene understanding, and mapping [4], particularly in unknown environments.

In the context of robot mapping, where robots are employed to incrementally generate map representations of an unknown environment, active perception enables autonomous mapping and plays a crucial role in improving both the efficiency and quality of the resulting map. This becomes even more important in resource-constrained online missions, where robots must operate under limited mission resources [96], such as operation time, number of measurements, or travel distance. By coupling perception and sensor control, the closed-loop setup in active perception allows the robot to make informed decisions about where to measure next, thereby reducing redundant measurements and focusing sensing efforts on the most task-relevant regions.

Active perception for robot mapping typically follows an iterative process that alternates between planning future sensor viewpoints, acquiring new measurements at those viewpoints, and updating the map representation accordingly. This cycle continues until the robot either completes the mission requirements, such as achieving full area coverage, or exhausts its available mission resources. Central to this loop is the view planning module, which selects future viewpoints based on the expected utility of the measurements they would provide. This allows the robot to act purposefully, directing its sensor viewpoints toward areas that are likely to yield more informative measurements to the mapping process, thereby enhancing the overall mapping efficiency and quality.

2.2.1 Utility Formulation

Most active perception approaches for robot mapping rely on explicit utility functions to guide view planning. Based on the current map state, these functions serve as a quantitative measure of the expected value of a potential measurement at candidate viewpoints with respect to specific mapping objectives. In general, the utility function can be formulated as $\psi : (\mathcal{M}, v) \rightarrow \mathbb{R}$, where $\mathcal{M}$ is the current map, and v is a candidate viewpoint, as illustrated in Figure 2.9. By assigning utility scores to candidate viewpoints, utility functions enable the robot to make informed choices, balancing between exploring new areas, refining uncertain regions, and revisiting known locations to improve map accuracy. A core contribution of this thesis lies in the formulation of utility functions tailored to various learning-based mapping techniques, which is essential for effectively integrating active perception with the used representations.

However, formulating utility functions in active perception for robot mapping

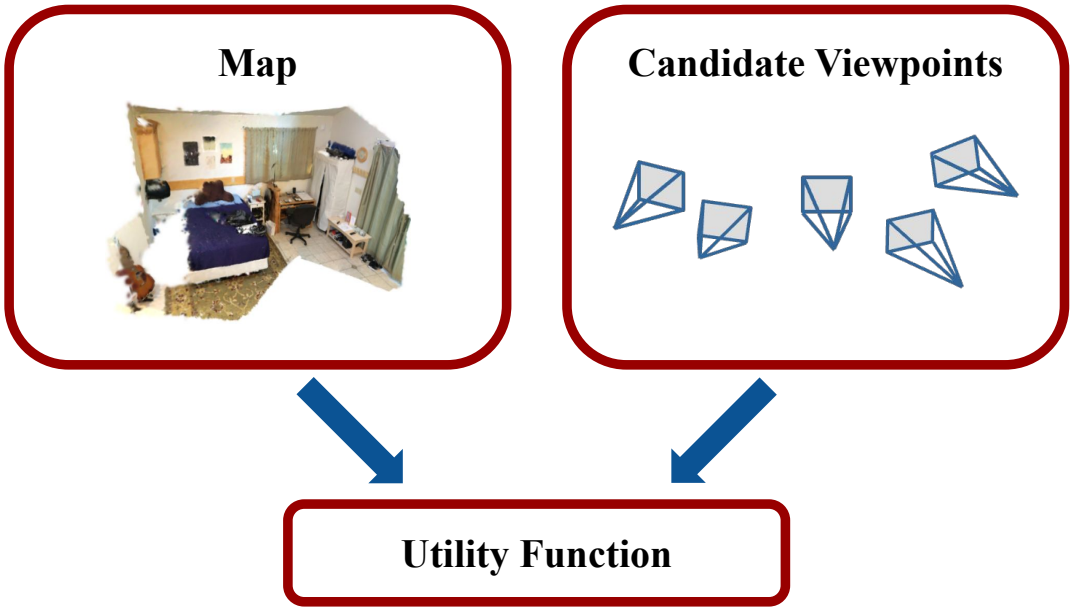


Figure 2.9: A general framework of utility-based view planning in active perception for robot mapping. The key in the view planning is a utility function, which reflects specific mapping objectives. Given a set of candidate viewpoints and the current map state, the utility function evaluates their expected utility values, indicating the potential contribution of new measurements at these viewpoints to the mapping objective.

can be non-trivial, as it is intricately linked with the choice of map representations tailored for specific applications, which in turn is heavily influenced by the objectives of the mapping task. For instance, in scene-level exploration tasks, the goal is to cover all unexplored areas in an unknown environment. Due to their capability to represent both surfaces and free space information, volumetric maps are often preferred for these tasks. In these scenarios, the mapping objectives are typically defined in terms of maximizing the coverage of the unknown environment; therefore, viewpoints that can observe larger unexplored areas in the volumetric map are preferred [27, 219]. When modeling the scene attributes probabilistically, such as occupancy probability, volumetric maps can also be used to quantify the uncertainty of the map from a novel viewpoint, allowing the robot to select viewpoints that minimize the uncertainty in the map [64, 122]. Similarly, mapping methods employing GPs inherently model map uncertainty through variance in Gaussian distributions [140], which can be directly incorporated into the utility function. Therefore, active perception for robot mapping utilizing GPs often relies on simulated measurements to identify viewpoints that are expected to reduce overall map uncertainty most effectively [54, 133].

For tasks that require the map to preserve fine-grained scene details, especially for object-level reconstruction or photorealistic rendering, map representations such as meshes, or more advanced learning-based representations like

NeRFs or GS maps, are often preferred. In these applications, the mapping objectives primarily focus on maximizing the geometric or textural fidelity of the reconstructed scene. Consequently, the utility function is typically designed to evaluate the expected quality of the map from a candidate viewpoint, such as the expected photometric loss or geometric error. A core challenge in such applications is the lack of ground-truth information at unseen viewpoints. As a result, it is difficult to directly compute the utility of a viewpoint beforehand. To address these, many approaches rely on heuristic estimates [51, 80, 161] or learning-based methods [89, 124, 204] to predict the utility values for viewpoint evaluation.

In summary, the utility function's design is pivotal in guiding view planning in active perception systems by aligning with specific map representations and mapping objectives, shaping how robots explore and reconstruct environments effectively. In this thesis, we propose utility functions designed for learning-based map representations, encompassing approaches that range from inherent uncertainty prediction and heuristic modeling to fully data-driven learning methods.

2.2.2 Candidate Viewpoint Generation

Beyond the design of the utility function itself, the strategy for generating candidate viewpoints is also critical in view planning. For generating candidate viewpoints, three major paradigms are commonly used: sampling-based, optimization-based, and heuristic-based methods.

Sampling-based methods work by generating a discrete set of candidate viewpoints either randomly or based on certain rules, such as weighted sampling to prioritize certain regions of the environment based on previous utility evaluation [10, 58, 71, 151, 186]. Sampling-based methods are generally easy to implement, efficient when utility evaluation is fast, and can be generalized to most of the map representations, including conventional and learning-based maps. The constraints of viewpoint space can easily be considered in the sampling step, making it flexible to adapt to different robot platforms and environments. However, their performance heavily depends on the quality and density of the candidate samples. For example, a large number of viewpoint samples may be required to cover the viewpoint space sufficiently, which can introduce redundancy and increase computational overhead, especially in large-scale or complex environments.

Optimization-based methods, on the other hand, formulate the view planning as an optimization problem, where the goal is to maximize the utility function over an action space of viewpoints [1, 203]. This approach often allows for more effective exploration of the action space and can yield better final utility values. It is particularly useful in scenarios where fine-grained control over the viewpoint is necessary. A critical requirement for optimization-based methods is that the utility function must be differentiable with respect to the viewpoint parameters,

which can be difficult to achieve for certain map representations or utility formulations. Both sampling-based and optimization-based methods may struggle to find informative viewpoints in scenarios where the utility function is highly non-linear or contains many local maxima, as they are prone to getting trapped in suboptimal solutions.

In contrast, heuristic-based methods, such as frontier-based exploration [16, 80, 202], often perform more robustly in such settings by leveraging domain-specific insights. For example, frontier-based methods generate candidate viewpoints directly in frontier regions, which are the boundaries between known and unknown space. The viewpoints in these regions are more likely to yield high utility in exploration tasks. While heuristic methods can efficiently guide the sensor viewpoints to promising areas without requiring extensive sampling or optimization, they typically rely on hand-crafted rules that are tailored to specific map representations. This dependence on manually designed heuristics can limit their generalization ability to different map representations.

In practice, some state-of-the-art systems adopt hybrid approaches that combine the strengths of these strategies. For instance, sampling-based methods may be used to generate a diverse set of initial candidate viewpoints, which are then refined through local optimization to reach viewpoints yielding higher utility values [203]. This combination enables both broad exploration of the viewpoint space and fine-tuned exploitation of promising regions, leading to more effective and efficient view planning in active perception for robot mapping. To further mitigate the risk of getting trapped in local maxima, heuristic-based methods can be integrated with sampling strategies to achieve local inspection via sampled candidates, while reserving the ability to explore a broader viewpoint space using heuristically generated candidates [80].

2.2.3 Viewpoint Selection Strategies

Besides utility formulation and candidate viewpoint generation, the strategy for selecting the next viewpoints also impacts the effectiveness of view planning. Without any prior knowledge, a widely used strategy is to greedily select the viewpoint with the highest expected utility from the candidate viewpoints, which is often referred to as the NBV planning [58, 132] and can be formulated as:

$$v^* = \arg\max_{v} \psi(\mathcal{M}, v) - \delta(\mathcal{M}, v, \hat{v}), \tag{2.19}$$

where δ can be an optional cost term, e.g., depending on the travel distance from the current viewpoint $\hat{v}$ to a candidate viewpoint v.

NBV planning is straightforward and computationally efficient; yet, due to its greedy nature, NBV planning tends to exhibit myopic behavior, prioritizing immediate utility gains without considering the long-term consequences of

viewpoint selection. This can lead to suboptimal exploration paths, such as a back-and-forth movement pattern, limiting the efficiency of robot mapping.

To address this limitation, receding horizon approaches offer a compromise between short-term efficiency and long-term effectiveness. Instead of only considering a single best viewpoint, receding horizon approaches evaluate short sequences of future viewpoints, computing the cumulative utility over each candidate sequence [12, 126, 175]. The robot then executes the first viewpoint from the most promising sequence and repeats the process in a rolling fashion. By incorporating a planning horizon, this strategy anticipates the downstream effects of its sensing actions, leading to more informed decisions over time compared to myopic NBV planning. However, receding horizon approaches can be computationally expensive and often require path planning algorithms involved to provide a valid sequence of viewpoints, e.g., using rapidly-exploring random trees [87].

Different from the utility-based view planning paradigm, another line of research explores the use of reinforcement learning to learn a policy for view planning [72, 86, 211]. These approaches implicitly embed the utility function into the learning objective, enabling the robot to learn policies that project the current map state to the next sensing action. Depending on the formulation, reinforcement learning methods either select the most promising viewpoint from a set of candidates or directly output control actions for sensor movement, e.g., move forward a certain distance or rotate by a certain angle. While reinforcement learning can yield efficient and adaptive view planning behaviors, these approaches typically require large amounts of training data and extensive simulation or real-world experience to converge. Moreover, they often suffer from limited generalization, performing poorly when deployed in unseen environments or when transferred across different map representations.

In this thesis, we develop active perception approaches for robot mapping that explicitly consider utility formulation for view planning. Our goal is to design utility functions that are specifically tailored to different learning-based map representations and mapping objectives. To guide the robot's exploration and measurement acquisition, we specifically focus on the sampling-based NBV planning strategy that balances efficiency and adaptability, enabling informed decision-making during online robot mapping.

Chapter 3

Adaptive-Resolution Field Mapping Using Gaussian Process Fusion with Integral Kernels

Robots have been widely adopted for mapping tasks due to their flexibility and high degree of autonomy. One important application scenario is environmental monitoring, which plays a central role in helping us understand the Earth and its natural processes. Many commonly observed natural phenomena, e.g., temperature and humidity, exhibit complex and non-uniform spatial variations that are difficult to capture using traditional monitoring methods [83], such as manual sampling or static sensor networks [101]. Recently, UAVs have emerged as a flexible, cost-efficient platform for measurement acquisition in a wide range of applications, including biomass calculation [134], signal strength monitoring [58], weed detection [163], and thermal mapping [101]. To fully exploit the autonomy of these platforms for environmental monitoring, particularly in the context of active perception for robot mapping, a key challenge is developing mapping approaches that can accurately capture heterogeneous natural phenomena while being compact and computationally efficient for online decision-making on resource-constrained robot platforms.

A variety of methods have been developed for field mapping in the context of environmental monitoring. In the remote sensing community, most existing approaches exploit aerial measurements to create high-resolution reconstructions, e.g., terrain orthomosaics [101]. Although they produce very detailed models, such procedures often require heavy offline postprocessing, making them unsuitable for online incremental mapping. To enable such applications, a common strategy is to discretize the environment in a grid map and fuse new measurements into it during a monitoring mission. However, traditional grid-based methods [35, 44, 59] assume spatial independence between cells, neglecting important

spatial correlations which characterize environmental phenomena, and thereby often limiting the map quality. In contrast, our goal is to develop an online mapping strategy that explicitly models spatial correlations by leveraging Gaussian processes (GPs) [140]. Our approach aims to support high-fidelity field reconstruction in targeted regions of interest, e.g., hotspots or anomalies, as well as online mapping with low computational and memory requirements. By addressing both fidelity and efficiency simultaneously, our work bridges the gap between environmental monitoring applications and the needs of active perception for robot mapping, where online reasoning and decision-making are essential capabilities.

This chapter focuses primarily on the mapping component of an active perception system, targeting the reconstruction of continuous, spatially correlated 2D scalar fields, e.g., of temperature or biomass cover, using measurement information from onboard sensors. We emphasize that mapping quality and efficiency could be critical factors for adaptive view planning in active perception systems, as they rely on current map states to inform the measurement acquisition process. This is particularly relevant when using GPs as the map representation, which offer built-in uncertainty modeling that can be directly utilized for view planning. However, view planning using GPs often requires forward simulation of map updates to select the most informative viewpoints, which can become a computational bottleneck in online settings. In this context, improving the efficiency and adaptability of the GP-based mapping directly enhances the robot's capability for intelligent decision-making in view planning. By reducing the computational burden of continuous field mapping using GPs, our method lays the foundation of active perception approaches for robot mapping that rely on accurate uncertainty estimates in GPs for viewpoint selection. Thus, although this chapter focuses on the mapping side, it forms a crucial component for active perception systems, where mapping quality and computational tractability are tightly coupled with view planning performance.

We follow GP fusion [133, 185] for online field mapping. In GP fusion, a GP model is exploited to initialize a spatially correlated grid map, which serves as a prior for recursive Bayesian fusion. Although discrete grid maps are used to store and update environmental information, the underlying GP allows continuous inference at arbitrary resolutions, as long as its probabilistic structure is preserved. The goal of our approach is to adaptively adjust the GP fusion map resolution online based on the information value of associated measurements, such that only regions of interest are mapped at high resolutions. This leads to compact map representations that are computationally efficient and memory-friendly during online mapping, while still preserving fine-grained details in regions of interest. Different from GP regression introduced in Chapter 2.1.1, which pools the entire measurement history to predict the posterior map state at any resolution at once,

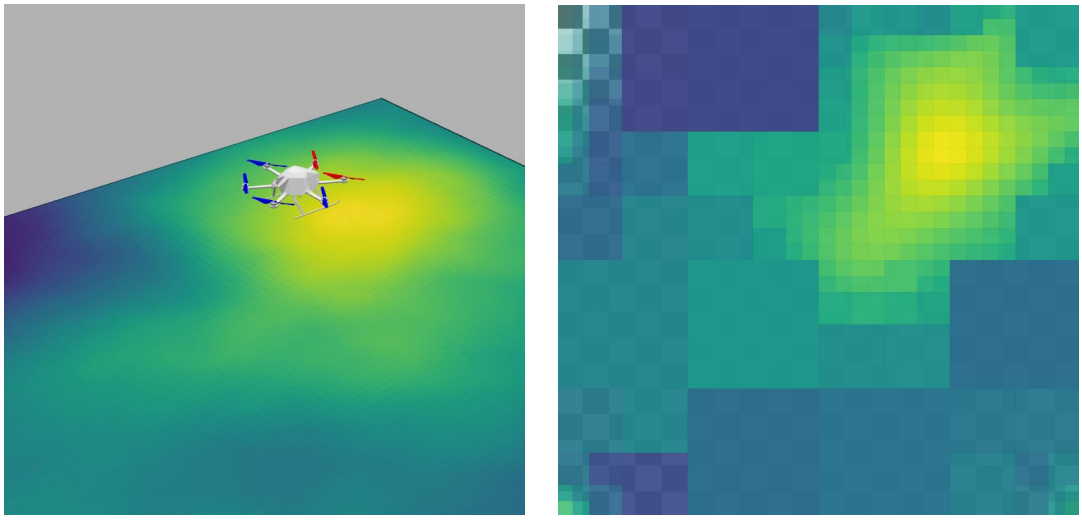


Figure 3.1: Our adaptive-resolution GP fusion approach for online field mapping. Left: Synthetic ground-truth distribution. Yellower shades indicate higher values we would like to map in greater detail. Right: Mapping result with uncertainty. Our approach maps regions of interest at higher resolutions while compressing information in less interesting regions to increase computational and memory efficiency. The checkerboard serves as an interpretation of map uncertainty (high opacity means low uncertainty).

the usage of GP fusion, although more efficient, poses a major challenge: adapting the map resolution leads to varying mapping locations in the environment; however, correlations at these new locations cannot be easily obtained from the previous measurements or the current map state [141]. Naively merging grid cells in GP fusion would lead to a loss of spatial correlations, as the posterior at the new resolution is not guaranteed to be consistent with the previous one. The posterior after map resolution change is thus difficult to retrieve in a theoretically sound and efficient manner. This hinders the recursive update step and constitutes an open research question.

To address this, we propose a novel GP formulation based on integral kernel functions to describe the spatial correlation over the areas of grid cells instead of points, e.g., grid cell center point. This area-based kernel formulation naturally introduces a hierarchical structure in the modeling of spatial correlations, enabling more efficient fusion of posterior information, as discussed later in Section 3.1. Combined with an Nd-tree structure [35], we adapt the map resolution online while preserving its spatial correlations. This enables us to retain high-resolution details in targeted areas of the field, while using coarser resolutions otherwise, as shown in Figure 3.1. In this way, we achieve memory and computationally efficient mapping without sacrificing map quality, as necessary for online applications on platforms, e.g., UAVs, with limited computing power.

We make the following three claims:

1. We propose an integral kernel formulation to encode the spatial correlations over the areas of 2D grid cells, enabling efficient merging operation of grid cells to compress information at any scale in uninteresting regions while preserving spatial correlations in the map.

2. Our approach combines GP fusion with the Nd-tree data structure to allow for resolution adaptation of spatially correlated maps, enhancing the mapping efficiency and reducing memory usage for online mapping compared to state-of-the-art baselines. We also demonstrate its applicability in a surface temperature mapping scenario.

3. We demonstrate the effectiveness of our mapping approach in active perception for robot mapping, justifying the high mapping quality and efficiency of our approach benefit online adaptive view planning.

3.1 Our Approach to Adaptive-Resolution Gaussian Process Fusion

This section introduces our online field mapping approach. We initialize a grid map using a GP model and store it in an Nd-tree. This map is then recursively updated with new measurements using Bayesian fusion. We first present the theory behind GPs with the integral kernel and define an average measurement sensor model, in which the state of a grid cell represents the average function value over the cell area. Then, we explain our Bayesian fusion update and the merging operation for incrementally building adaptive-resolution field maps. Bringing together these elements, our key contribution is the ability to efficiently merge grid cells in GP fusion without losing spatial correlations. Note that our setting considers a UAV-based mapping scenario. However, our approach is applicable to general 2.5D mapping problems.

3.1.1 Gaussian Processes and Integral Kernels

A GP is the generalization of a Gaussian distribution over a finite vector space to an infinite-dimensional function space. It is fully described by its mean $\mu(\mathbf{x})$ and kernel function $k(\mathbf{x}, \mathbf{x}')$, where $\mathbf{x}$ is an arbitrary point in input space. In practice, a GP regression model is used to encode spatial correlations in a probabilistic non-parametric manner and infer function values at a finite set of query points given observed measurements [140]. Different from GP regression, previous studies of GP fusion [133, 185] exploit the GP's mean and kernel function to calculate the prior in predefined mapping positions, e.g., grid cell center points. The posteriors at these points are then recursively updated with grid cell measurements

using Bayesian fusion, assuming measurements falling into the same grid cell as direct measurements of the grid cell center point. This GP fusion setting largely enhances the map update efficiency compared to standard GP regression. However, as the map posterior is only maintained in fixed mapping positions, adaptive resolution is hard to achieve.

To address this problem, we propose a new GP fusion approach leveraging an integral kernel. The mapping target in our problem is assumed to be a stationary continuous function described by a GP: $f(\mathbf{x}) \sim \mathcal{GP}(\mu, k) : \mathcal{E} \to \mathbb{R}$, where $\mathcal{E} \subset \mathbb{R}^2$ is the 2D rectangular input space and $\mathbf{x} \in \mathcal{E}$. Similar to Reid *et al.* [142], we now define the new function:

$$\zeta(r) = \frac{1}{a} \int_r f(\mathbf{x}) \, \mathrm{d}\mathbf{x}, \tag{3.1}$$

to represent the average of the latent function f over a rectangular domain $r \subset \mathcal{E}$ with area $a \in \mathbb{R}$. Since applying a linear operator to a GP leads to another GP [148], we obtain the new GP: $\zeta(r) \sim \mathcal{GP}(\mu_\mathrm{I}, k_\mathrm{I})$, whose mean and kernel function are described as follows:

$$\mu_\mathrm{I}(r_i) = \frac{1}{a_i} \int_{r_i} \mu(\mathbf{x}) \, \mathrm{d}\mathbf{x}, \tag{3.2}$$

$$k_\mathrm{I}(r_i, r_j) = \frac{1}{a_i a_j} \iint_{r_i \times r_j} k(\mathbf{x}, \, \mathbf{x}') \, \mathrm{d}\mathbf{x} \, \mathrm{d}\mathbf{x}', \tag{3.3}$$

where $\mathbf{x}$ and $\mathbf{x}'$ are the point positions contained within the rectangular domains r_i with area a_i and r_j with area a_j respectively. The area-related terms in Equation (3.2) and Equation (3.3) simply transform the integral into an average, which makes the physical meaning of mean and covariance in accordance with our measurement model introduced in Section 3.1.3. For simplicity, we consider a constant mean function μ in our approach.

3.1.2 Map Initialization

We initialize our grid map using this new GP model. For rectangular cells and squared exponential kernel [140]:

$$k(\mathbf{x}, \, \mathbf{x}') = \sigma^2 \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|_2^2}{2\ell^2} \right), \tag{3.4}$$

we can find a closed-form solution to Equation (3.3). In general, numerical integration is required to determine the kernel integration [116]. Note that the integral calculation is only conducted in the initialization step and does not burden online mapping. The fact that our model is a GP allows us to initialize the prior map at any resolution, and we recursively discretize the input space into rectangular grid cells using an Nd-tree until maximum depth $t \in \mathbb{N}^+$

is reached. Only leaf grid cells are maintained and updated in our grid map $\mathcal{C} = \{c_1, \ldots, c_n\}$, where $n = (N^d)^t$ with $d = 2$, as we focus on 2D field mapping; $c_i = [x_i^{\min}, x_i^{\max}] \times [y_i^{\min}, y_i^{\max}]$ is the parametrization of a grid cell $c_i \subset \mathcal{E}$, and $\mathbf{c} = [c_1, \ldots, c_n]^\top$ is the vectorization of $\mathcal{C}$. This prior map, with prior mean vector $\boldsymbol{\mu}^-$ and covariance matrix $\mathbf{K}^-$ calculated by Equation (3.2) and Equation (3.3):

$$\boldsymbol{\mu}^- = \mu_{\mathrm{I}}(\mathbf{c}) = \begin{bmatrix} \mu_{\mathrm{I}}(c_1) \\ \vdots \\ \mu_{\mathrm{I}}(c_n) \end{bmatrix}, \mathbf{K}^- = k_{\mathrm{I}}(\mathbf{c}, \mathbf{c}) = \begin{bmatrix} k_{\mathrm{I}}(c_1, c_1) & \ldots & k_{\mathrm{I}}(c_1, c_n) \\ \vdots & \ddots & \vdots \\ k_{\mathrm{I}}(c_n, c_1) & \ldots & k_{\mathrm{I}}(c_n, c_n) \end{bmatrix}, \quad (3.5)$$

can be seen as a multivariate Gaussian distribution from the perspective of the recursive update introduced later in Section 3.1.4:

In our online mapping approach, we initialize the map to the highest resolution and adaptively merge uninteresting grid cells during mapping. Note that, since our underlying model is a GP, we can still infer the function values at arbitrary resolutions using the mean function, kernel function, and map posteriors as described by Reece *et al.* [141]. As this procedure is computationally heavy, we only consider it as a post-processing step to recover a high-resolution map after an online mission is complete.

3.1.3 Sensor Model

In our GP fusion setting, we consider a Gaussian sensor model to account for noisy measurements. For each observed grid cell $c_i \in \mathcal{C}$, the sensor provides a measurement y_i capturing the average value of function f over the area of this cell as $y_i \sim \mathcal{N}(\mu_{s,i}, \sigma_{s,i}^2)$, where $\mu_{s,i}$ is the mean and $\sigma_{s,i}^2$ is the variance expressing uncertainty in y_i. The variance can be further decomposed into two parts. First, we assume measurements taken from higher altitudes are more susceptible to environmental noise. To this end, we follow the work of Popović *et al.* [133] and describe the degraded accuracy of sensor information at higher altitudes by $\sigma_{a,i}^2 = \alpha h$, where $\alpha \in \mathbb{R}^+$ is a coefficient and h is the sensor's altitude. Second, we consider uncertainty caused by observing incomplete grid cells. In our mapping approach, some grid cells are only partially covered by the current sensor footprint, especially when the grid cells occupy larger area after they are merged. Directly assigning the average values as the final measurement of these grid cells would be an over-confident assumption, as the unobserved part of these grid cells may contradict the current measurements, e.g., when grid cells span over the domain of heterogeneous function values. To tackle this problem, we propose the coverage-ratio-dependent variance $\sigma_{c,i}^2 = \beta \left(1 - \frac{a_{\mathrm{cover}}}{a_c}\right)$ in our sensor model, where $\beta \in \mathbb{R}^+$ is a coefficient, and a_c, a_{cover} are the area of the grid cell and the part covered by the sensor footprint.

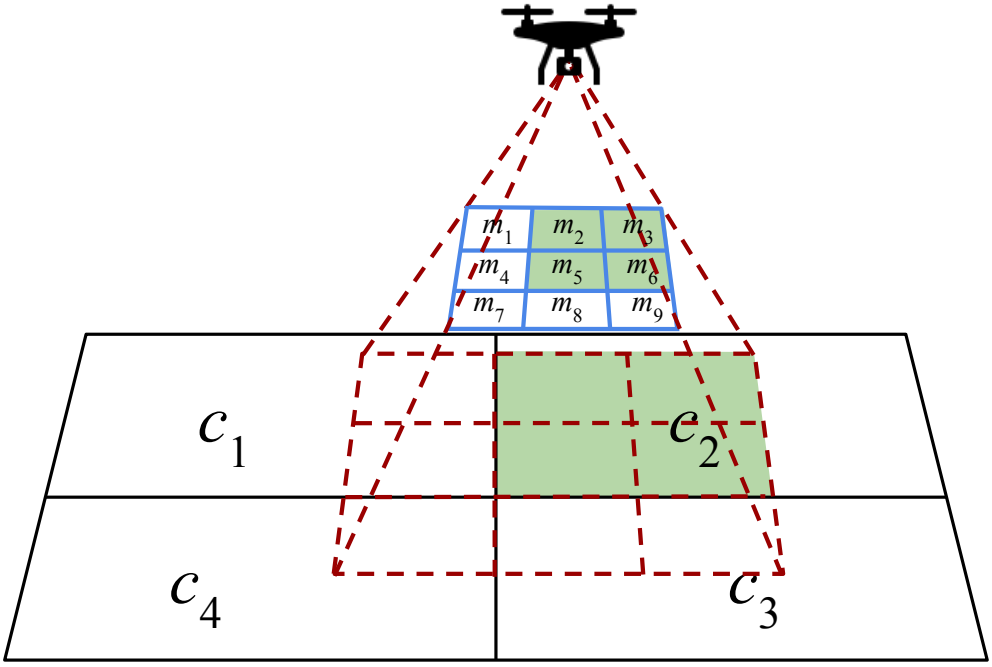


Figure 3.2: Our sensor model provides the measurements of the average function value over a grid cell. For instance, the measurement z_2 observed from c_2 is the average of four single measurement values $\{m_2, m_3, m_5, m_6\}$. For calculating $\sigma^2_{c,i}$, a_{cover} is the green area on the terrain and a_c is the area of c_2 itself.

For each new measurement, the data are generated as follows. First, the sensor footprint is determined based on the known field of view and the sensor's extrinsic parameters. Next, we identify the grid cells having overlap with the sensor footprint using a depth-first tree search with pruning. For each observed grid cell c_i, we calculate the corresponding averaged measurement value y_i as illustrated in Figure 3.2. Finally, we sum $\sigma^2_{a,i}$ and $\sigma^2_{c,i}$ as the total variance of each measurement y_i.

3.1.4 Sequential Map Update

A major difference between GP regression and our GP fusion approach lies in the map update rule. During the online mapping mission, the map state is fully described by the mean vector $\boldsymbol{\mu}$ and covariance matrix $\mathbf{K}$, which is initialized by our GP model as introduced in Chapter 3.1.2, and recursively updated by Bayesian fusion with new measurements. Specifically, we denote $\boldsymbol{\mu}^-$, $\mathbf{K}^-$ as the prior map state and $\boldsymbol{\mu}^+$, $\mathbf{K}^+$ as the posterior map state for map updates.

At each update step, $\mathbf{y}$ denotes a vector of l new average function value measurements observed from l corresponding grid cells, as introduced in Section 3.1.3. The posterior density $p(\boldsymbol{\zeta}|\mathbf{y}, \mathbf{c}) \propto p(\mathbf{y}|\boldsymbol{\zeta}, \mathbf{c})p(\boldsymbol{\zeta}|\mathbf{c})$, where $\boldsymbol{\zeta} = [\zeta(c_1), \ldots, \zeta(c_n)]$,

can be computed using the Kalman Filter update equations [141]:

$$\boldsymbol{\mu}^{+} = \boldsymbol{\mu}^{-} + \boldsymbol{\Gamma}\mathbf{g}, \tag{3.6}$$

$$\mathbf{K}^{+} = \mathbf{K}^{-} - \boldsymbol{\Gamma}\mathbf{H}\mathbf{K}^{-}, \tag{3.7}$$

where $\boldsymbol{\Gamma} = \mathbf{K}^{-}\mathbf{H}^{\top}\mathbf{S}^{-1}$ is the Kalman gain; $\mathbf{g} = \mathbf{y} - \mathbf{H}\boldsymbol{\mu}^{-}$ and $\mathbf{S} = \mathbf{H}\mathbf{K}^{-}\mathbf{H}^{\top} + \mathbf{R}$ are the measurement and covariance innovations; $\mathbf{R}$ is a diagonal $l \times l$ matrix composed of variance term $\sigma_{a,i}^2 + \sigma_{c,i}^2$ associated with each measurement y_i and $\mathbf{H}$ is a $l \times n$ observation matrix denoting the part of the map observed by $\mathbf{y}$, where n and l are the number of grid cells in the current map and observed grid cells, respectively. Note that the current map only contains leaf grid cells and a small matrix $\mathbf{S} \in \mathbb{R}^{l \times l}$ is inverted at each update.

3.1.5 Merging Operation

Given a non-uniform target field for mapping, our goal is to use coarser (larger) grid cells to map uninteresting regions and denser (smaller) grid cells to retain details in interesting parts. Previous works utilizing GP fusion [133, 185] do not support efficient resolution changes. By using our new GP fusion method with the integral kernel k_{I}, however, we naturally encode the states of parent nodes in their children, which enables efficient retrieval of a parent's posterior from its children on the fly.

The online merging operation allows us to summarize information in larger areas and monotonically reduce the total number of grid cells in the map, which facilitates mapping efficiency and memory usage. For this, we subdivide our map into uninteresting regions (UR) and regions of interest as hotspots (HS):

$$\mathcal{C}_{UR} = \{c_i \in \mathcal{C} \mid \boldsymbol{\mu}_i + \gamma \mathbf{K}_{i,i} \leq f_{th}\},\ \mathcal{C}_{HS} = \mathcal{C} \setminus \mathcal{C}_{UR}, \tag{3.8}$$

where $\boldsymbol{\mu}_i$ and $\mathbf{K}_{i,i}$ are the mean and variance of grid cell c_i in the current map; the design parameter γ is chosen to specify the margin to the threshold f_{th} [55]. The threshold f_{th} can be defined by expert knowledge in a certain application, e.g., in agricultural scenarios, high temperature may indicate crop health issues and thus be more interesting for environmental monitoring tasks. We also consider the variance information to avoid merging grid cells with possibly high mean values, which causes loss of details in interesting regions. Note that Equation (3.8) can be easily rewritten to account for interesting regions with low mean values.

For a parent grid cell, if all of its $P = N^d$ child grid cells are uninteresting leaves (grid cells in $\mathcal{C}_{UR}$), these child grid cells can be replaced by their parent grid cell. When we merge the information of P children into their parent, based on the definition of the grid cell variable and the correlation encoded by the integral kernel, we have the parent grid cell defined as $\zeta_{\mathrm{parent}} = \frac{1}{P}\sum_{i=1}^{P} \zeta_{\mathrm{child}_i}$. The parent

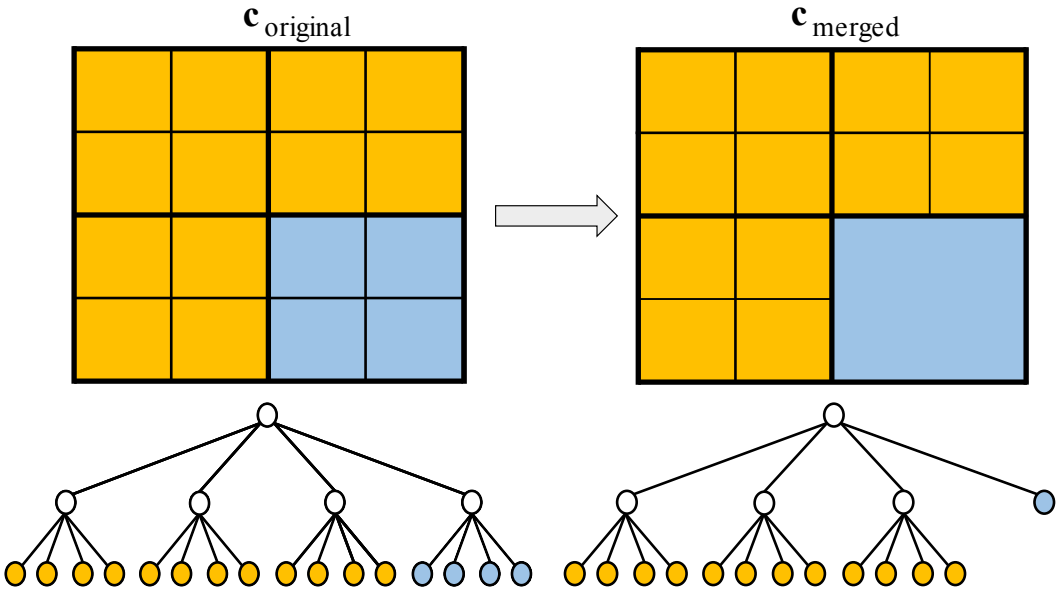


Figure 3.3: Illustration of the merging operation in our map. Top and bottom rows show the grid cell map and its corresponding Nd-tree (with $N = d = 2$) structure. Only leaf nodes (blue and orange) are considered in the map update. After merging (right), the states of child grid cells are summarized into their parent in the new map.

grid cell now represents the average function value of the entire region covered by its children. For the grid map, the merging operation can be described as the linear transformation of a multivariate Gaussian distribution:

$$\boldsymbol{\mu}_{\text{merged}} = \mathbf{M}\boldsymbol{\mu}_{\text{original}}, \tag{3.9}$$

$$\mathbf{K}_{\text{merged}} = \mathbf{M}\mathbf{K}_{\text{original}}\mathbf{M}^{\top}, \tag{3.10}$$

where $\boldsymbol{\mu}_{\text{original}}$, $\mathbf{K}_{\text{original}}$ represent the map state of $\mathbf{c}_{\text{original}}$ prior to merging, and $\boldsymbol{\mu}_{\text{merged}}$, $\mathbf{K}_{\text{merged}}$ denote the map state of the newly-merged map $\mathbf{c}_{\text{merged}}$. In the simplest case, where only one parent's child cells are merged, $\mathbf{c}_{\text{orginal}}$, $\mathbf{c}_{\text{merged}}$, and $\mathbf{M}$ can be expressed as:

$$\mathbf{c}_{\text{original}} = \begin{bmatrix} c_1 \\ \vdots \\ c_{n-P} \\ c_{n-P+1} \\ \vdots \\ c_n \end{bmatrix}, \qquad \mathbf{c}_{\text{merged}} = \begin{bmatrix} c_1 \\ \vdots \\ c_{n-P} \\ c_{n-P+1} \end{bmatrix}, \tag{3.11}$$

$$\mathbf{M} = \begin{bmatrix} \mathbf{I}_{(n-P)\times(n-P)} & \mathbf{0}_{(n-P)\times P} \\ \mathbf{0}_{1\times(n-P)} & \mathbf{q}_{1\times P} \end{bmatrix}, \tag{3.12}$$

assuming that c_{n-P+1} in $\mathbf{c}_{\text{merged}}$ now represents the parent of grid cell to be merged $\{c_{n-P+1}, \ldots, c_n\}$ in $\mathbf{c}_{\text{original}}$, and $\mathbf{q}$ is $\left[\frac{1}{P}, \ldots, \frac{1}{P}\right]$. A simple illustration is given in Figure 3.3. The merging operation is performed for eligible grid cells after every map update. As the multivariate Gaussian distribution is closed under linear transformations, the map after the merging operation becomes the prior map for the Bayesian fusion update described in Chapter 3.1.4.

3.2 Experimental Evaluation

Our experimental results support our three claims: (i) we show that our integral kernel-based formulation enables merging operations to compact the map resolution in uninteresting areas; (ii) we show that our approach achieves on-pair mapping accuracy while providing more efficient map updates and reduced memory usage compared to various baselines for field mapping; and (iii) we further integrate our adaptive-resolution mapping approach with view planning to demonstrate that it benefits active perception for robot mapping.

3.2.1 Mapping Evaluation

We evaluate the mapping performance with total mapping time, mapping quality in terms of root mean square error (RMSE), intersection over union (IoU) of hotspots, memory consumption ratio, and number of grid cells in the final maps. The total mapping time is obtained by aggregating the individual map update times over the mission; RMSE and IoU are calculated by comparing resulting maps and ground truth at the ground-truth resolution; memory usage is reported as a ratio relative to the approach with the highest memory consumption. We compare six different mapping approaches:

- *Ours*: our adaptive-resolution mapping strategy based on GP fusion with integral kernel as described in Section 3.1;

- *FR-IDP*: fixed-resolution mapping under independence assumption [36]. We use the same map initialization and update strategy as introduced in Section 3.1, while setting the non-diagonal elements of the covariance matrix to zero, meaning that the spatial correlations between grid cells are not considered for map updates;

- *AR-IDP*: adaptive-resolution mapping under independence assumption. Uninteresting grid cells are pruned during mapping as proposed by Einhorn *et al.* [35];

- *AR-BCM*: adaptive-resolution mapping using the Bayesian committee machine [181] and test-data tree, as adapted from Wang *et al.* [192]. Uninteresting grid cells are pruned to reduce the number of query points. We do not follow the nested Bayesian committee machine approach, as our whole map can be seen as a block in their case;

- *AR-GPR-IK*: adaptive-resolution GP regression with integral kernel based on the approach proposed by Reid *et al.* [142]. We take one step further to recursively merge cells if they are uninteresting after each regression update;

- *FR-GPF*: fixed-resolution GP fusion proposed by Popvić *et al.* [134].

We simulate 20 different $20\,\mathrm{m} \times 20\,\mathrm{m}$ Gaussian random fields with 400×400 resolution as ground-truth environments, representing spatially correlated variables over the terrain. For simplicity, we normalize the ground-truth values between $[0, 1]$ and define regions with values greater than 0.7 as hotspots. To model noisy measurements, we add zero-mean Gaussian noise to the ground-truth values, following the altitude-dependent noise model introduced in Section 3.1.3 with $\alpha = 0.03$. To assess mapping performance at different scales, we conduct experiments at 3 different maximum resolutions: 16×16, 32×32, and 64×64 grid cell maps corresponding to adaptive-resolution approaches with maximum quadtree depths of 4, 5, and 6, respectively.

We map the terrains using a lawnmower pattern to focus on comparing the mapping performance, excluding the influence of path variations. To simulate a UAV mission, we take 16 non-overlapping measurements as shown in Figure 3.4(a) to fully cover the terrain, assuming a flight altitude of 2.5 m and $5\,\mathrm{m} \times 5\,\mathrm{m}$ sensor footprint on the ground. All GP-based mapping approaches (*AR-BCM*, *AR-GPR-IK*, *FR-GPF*, *Ours*) use the squared exponential kernel function with hyperparameters $\{\sigma^2, \ell\} = \{1, 2.36\}$ and a constant prior mean value of 0.5.

In general, the domain knowledge should be exploited in kernel function selection, and the hyperparameters can be optimized using prior information, e.g., datasets from an earlier sampling campaign or similar fields, as discussed in Chapter 2.1.1. For approaches using an integral kernel (*AR-GPR-IK*, *Ours*), we follow Equation (3.5) to calculate the prior maps. For mapping under the independence assumption (*FR-IDP*, *AR-IDP*), we use the same prior mean and variance, while ignoring all cross-correlations to isolate each grid cell. We consider the average measurement sensor model in all mapping approaches. For merging operation in adaptive-resolution approaches, we choose $\{\gamma, f_{\mathrm{th}}\} = \{2, 0.7\}$ in Equation (3.8). Note that all these hyperparameters are used consistently in all experiments.

We summarize the results in Table 3.1 and Figure 3.4. In all cases, approaches relying on the cell independence assumption yield the least accurate maps with

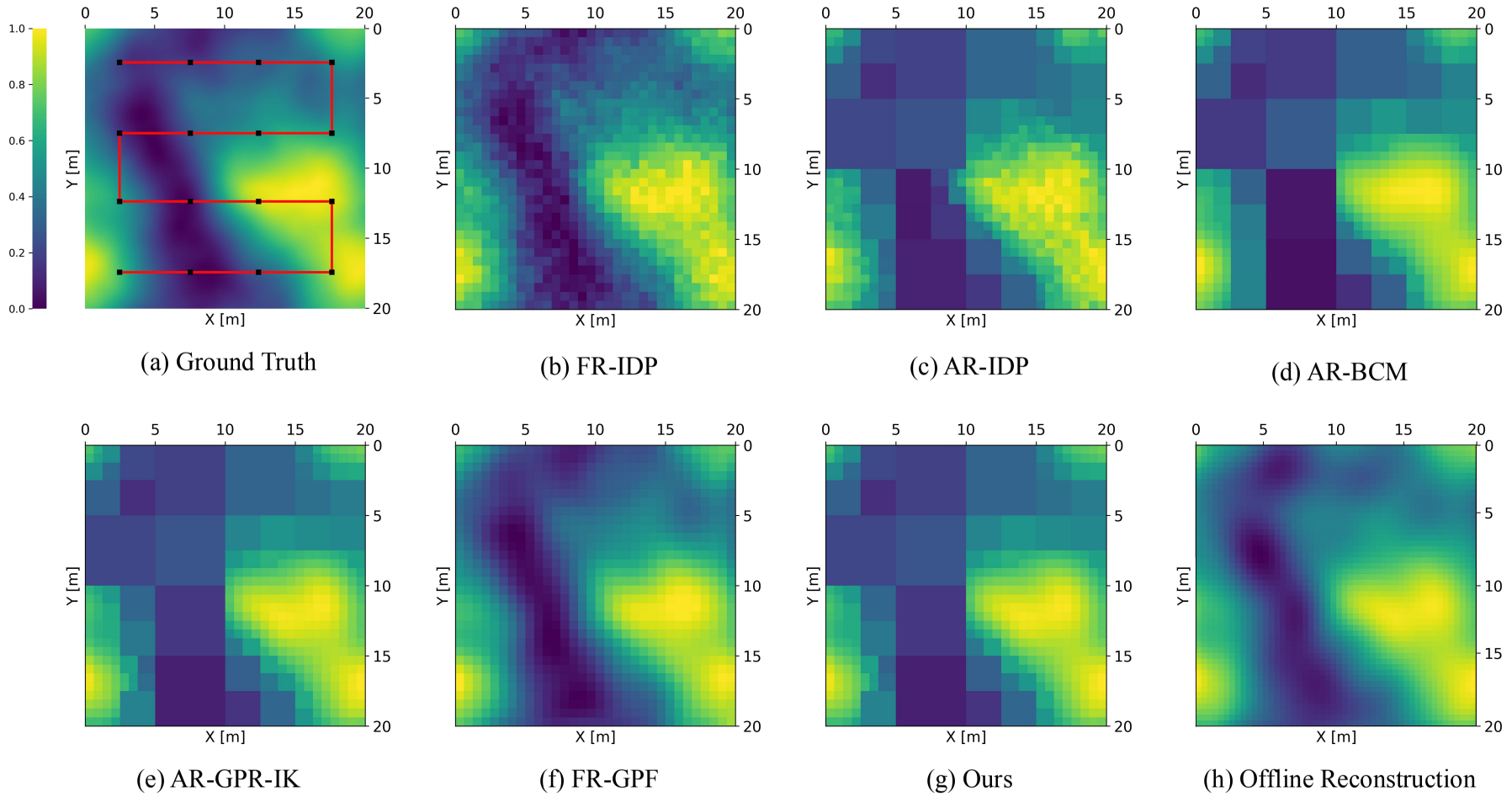


Figure 3.4: Qualitative comparison of our approach (g) against benchmarks (b)-(f). The terrain is mapped using a lawnmower pattern, as shown in (a). The red line and black dots indicate the traveled path and measurement locations. All approaches use a map size of 32×32 grid cells. By mapping adaptively, our method compresses information in areas with low information value (blue) while preserving details in higher-value regions of interest (yellow) to achieve a compact map representation for online applications. (h) shows the offline higher-resolution (50×50) reconstruction from our online mapping result (g), illustrating how the map can be decompressed after the mission.

Map size	Method	RMSE ↓	RMSE (hotspots) ↓	IoU (hotspots) ↑	Mapping time [ms] ↓	Memory usage ratio [%] ↓	Number of map cells ↓
16×16	FR-IDP	0.045 ± 0.002	0.045 ± 0.002	0.813 ± 0.024	5.575 ± 0.466	4.187 ± 0	256 ± 0
	AR-IDP	0.071 ± 0.003	0.046 ± 0.002	0.812 ± 0.024	7.299 ± 0.901	2.155 ± 1.127	125.2 ± 18.258
	AR-BMC	0.071 ± 0.003	0.038 ± 0.003	0.856 ± 0.023	273.546 ± 69.549	49.892 ± 11.433	115.22 ± 16.167
	AR-GPR-IK	0.065 ± 0.003	0.037 ± 0.002	0.856 ± 0.024	59.379 ± 20.680	40.692 ± 8.282	118.62 ± 16.907
	FR-GPF	0.037 ± 0.002	0.037 ± 0.002	0.857 ± 0.023	11.375 ± 0.965	100 ± 0	256 ± 0
	Ours	0.065 ± 0.003	0.037 ± 0.002	0.857 ± 0.026	10.168 ± 1.119	38.729 ± 9.416	114.4 ± 19.754
32×32	FR-IDP	0.065 ± 0.004	0.065 ± 0.004	0.723 ± 0.021	28.968 ± 0.834	1.067 ± 0	1024 ± 0
	AR-IDP	0.079 ± 0.005	0.067 ± 0.003	0.725 ± 0.020	63.796 ± 6.880	0.529 ± 0.068	508 ± 65.121
	AR-BMC	0.071 ± 0.005	0.025 ± 0.003	0.864 ± 0.023	13100.225 ± 2009.709	26.433 ± 5.359	356.25 ± 74.012
	AR-GPR-IK	0.066 ± 0.004	0.026 ± 0.003	0.866 ± 0.024	1747.631 ± 677.670	17.538 ± 5.640	371.3 ± 74.279
	FR-GPF	0.027 ± 0.002	0.026 ± 0.002	0.867 ± 0.024	430.843 ± 7.447	100 ± 0	1024 ± 0
	Ours	0.065 ± 0.004	0.026 ± 0.003	0.867 ± 0.025	261.763 ± 24.443	16.632 ± 5.112	360.4 ± 71.003
64×64	FR-IDP	0.123 ± 0.003	0.123 ± 0.008	0.625 ± 0.056	123.562 ± 4.271	0.268 ± 0	4096 ± 0
	AR-IDP	0.127 ± 0.004	0.124 ± 0.008	0.623 ± 0.058	586.427 ± 43.362	0.218 ± 0.023	2687.14 ± 383.364
	AR-BMC	0.104 ± 0.004	0.025 ± 0.003	0.869 ± 0.028	68645.365 ± 4606.239	15.782 ± 4.125	1258.5 ± 473.78
	AR-GPR-IK	0.073 ± 0.004	0.025 ± 0.003	0.872 ± 0.021	18620.558 ± 6448.663	11.552 ± 4.221	1387 ± 443.744
	FR-GPF	0.024 ± 0.002	0.024 ± 0.002	0.875 ± 0.023	9977.749 ± 251.003	100 ± 0	4096 ± 0
	Ours	0.073 ± 0.004	0.024 ± 0.002	0.876 ± 0.023	4098.029 ± 548.881	8.877 ± 4.824	1271.25 ± 484.191

Table 3.1: Comparison of our approach against baselines for varying map sizes. By combining GP fusion and adaptive-resolution mapping using an integral kernel, our strategy reduces runtime and memory consumption while delivering highly accurate maps.

the highest RMSE and the lowest IoU, since they are most vulnerable to sensor noise or sparse measurements. This is because they neglect correlations for mapping, which are key for capturing spatially correlated variables. In contrast, the four GP-based approaches reflect the smooth structure of the Gaussian random fields, as they incorporate covariance information into the map updates. As expected, the averaging effect caused by merging cells in adaptive-resolution approaches leads to higher total RMSE compared to *FR-GPF*. However, all GP-based approaches show comparable accuracy in mapping hotspots and similar IoU scores, as required in our problem setup.

In terms of mapping efficiency, *AR-BCM* performs the worst as it executes large matrix inversion and Bayesian committee machine fusion at every update step, leading to prohibitively slow mapping. Note that the Bayesian committee machine benefits from parallelizing several GP regressions. However, in online mapping scenarios, where measurements are accumulated incrementally, the Bayesian committee machine loses this strength. *AR-GPR-IK* is slower than two GP fusion approaches (*FR-GPF* and *Ours*), due to regression using accumulated measurements. We point out that by using the integral kernel together with the average measurement sensor model, *AR-GPR-IK* already achieves a significant speed-up compared to vanilla GP regression. In all cases, *AR-IDP* is slower than *FR-IDP* due to the overhead caused by tree search. The same overhead is expected in our approach; however, as the major bottleneck is the matrix inversion and multiplication in Equation (3.6) and Equation (3.7), this can be compensated by faster Bayesian fusion update with fewer grid cells in our approach.

Regarding memory usage, *FR-GPF* consumes the most memory space as it maintains a large constant number of grid cells and a large covariance matrix. Among the adaptive-resolution approaches, *AR-IDP* shows the worst merging ability, as indicated by the number of grid cells in the final map. This can be explained by heterogeneous states in children's nodes caused by inaccurate mapping, which potentially reduces the chances of the merging operation. Among the GP-based methods, our approach achieves the fastest mapping updates and best memory compression ratios with competitive map quality. The benefit of our online merging operation can be seen by comparing *Ours* and *FR-GPF*. In all cases, our approach outperforms *AR-IDP* and *FR-IDP* in terms of map quality. In Figure 3.4(h), we further show how our mapping result can be decompressed to recover a high-resolution reconstruction in an offline post-processing step, thanks to well-maintained spatial correlations in our map.

3.2.2 Validation on Real-World Data

We demonstrate our mapping approach in a real-world surface temperature mapping scenario. We collected sensor measurements in a $150\,\mathrm{m} \times 150\,\mathrm{m}$ crop field

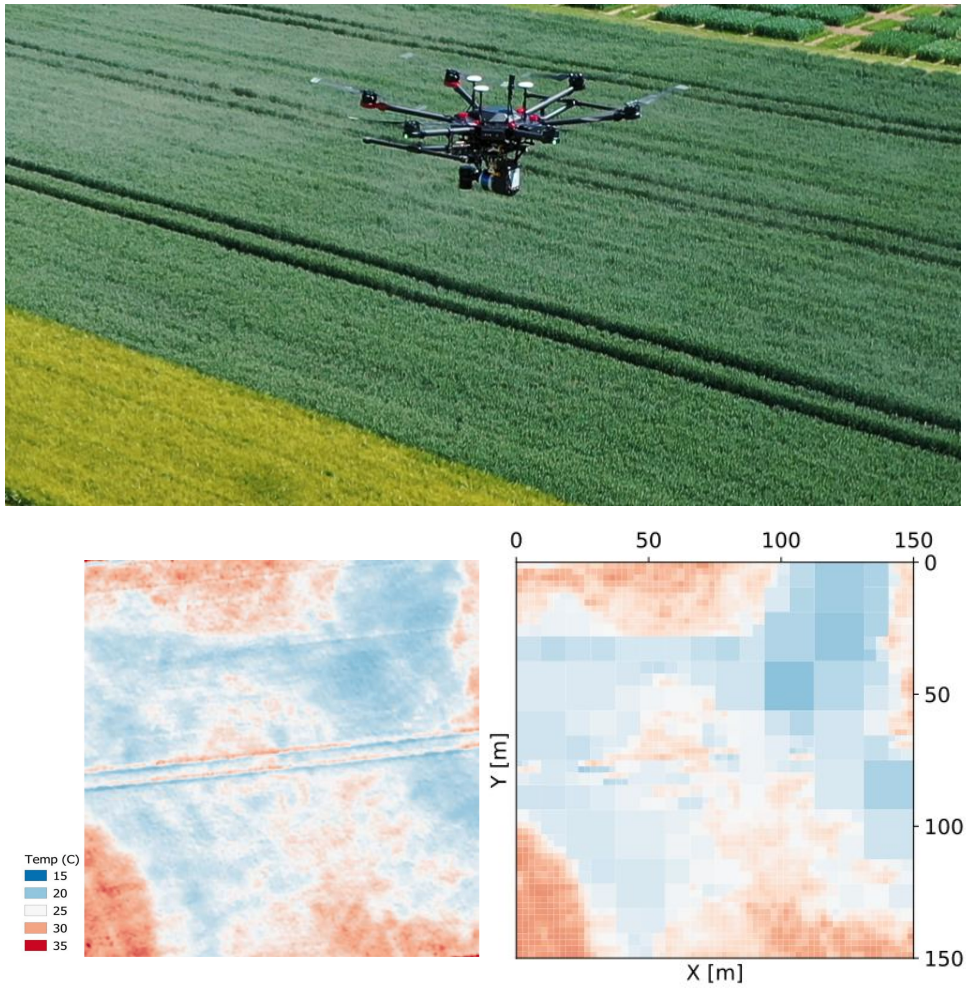


Figure 3.5: Validation of our approach for surface temperature mapping. Top: Experimental setup showing our UAV over a crop field. Bottom-left: Orthomosaic of the crop field's temperature distribution. Bottom-right: Map generated by our method. High-temperature areas (red) are mapped at higher resolutions to preserve detail in these regions of interest, while the grid cells in uninteresting regions are merged to compact the map representation, leading to more efficient map updates.

(50.86° lat., 6.45° lon.) near Jülich, Germany on June 25, 2021 using a DJI Matrice 600 UAV platform equipped with a Vue Pro R 640 thermal sensor. During measurement acquisition, the UAV followed a lawnmower path at 100 m altitude to collect thermal images at 15 cm ground resolution. The images were processed using Pix4D software to create an orthomosaic used as a proxy for ground truth in our experiment. We use a maximum map resolution of 64×64. The entire mapping process takes 28.31 s considering 81 measurements with 50% overlap. The aim is to validate our method for adaptively mapping hotspots ($> 28°$ C) at finer resolutions using these real measurements. The mapping result in Figure 3.5 confirms that our approach can adapt the map resolution in a targeted way.

3.2.3 Integration with Active Perception

Finally, to demonstrate that our proposed mapping approach can enhance active perception based on GPs, we integrate our mapping method with adaptive view planning to actively map 2D fields in UAV-based environmental monitoring missions, following the setting introduced by Popović *et al.* [133]. The planning task aims at efficient detection of regions of interest in an initially unknown environment under mission time constraints. For this, the UAV must adaptively plan its viewpoints based on the current map state to trade off between exploration of unknown areas and exploitation of already identified regions of interest.

This experiment considers the same setup as described in Section 3.2.1 except setting our prior mean to 0.7 to initially encourage exploration. We compare the *FR-IDP*, *AR-IDP*, *FR-GPF* methods to our approach, as regression-based mapping approaches *AR-BCM* and *AR-GPR-IK* are prohibitively slow for online planning. For all methods, we employ the same planning strategies to isolate the influence of mapping on planning performance.

We use a 3D lattice consisting of 300 total viewpoints at altitudes of 2 m and 5 m to represent the discrete action space. At each planning step, the UAV is allowed to move to one of these predefined viewpoints, each associated with a nadir-facing camera orientation. The planner applies greedy search among these candidate viewpoints to find the next best viewpoint by forward-simulating the map updates and calculating the expected reward for each. The utility function ψ is defined by the posterior variance reduction in regions of interest, assuming a measurement taken at a candidate viewpoint v:

$$\psi(v, \boldsymbol{\mu}^-, \mathbf{K}^-) = \sum_{c_i \in \mathcal{C}_{HS}} (\mathbf{K}_{i,i}{}^- - \mathbf{K}_{i,i}{}^+), \tag{3.13}$$

where $\mathcal{C}_{HS}$ is the hotspot area in the prior map as defined in Equation (3.8); $\mathbf{K}_{i,i}{}^-$ is the prior variance and $\mathbf{K}_{i,i}{}^+$ is the posterior variance of grid cell c_i after a simulated measurement at v, calculated by Equation (3.7). Taking the flight time cost into consideration, we finally get the next best viewpoint v^* by:

$$v^* = \arg\max_v \frac{\psi(v, \boldsymbol{\mu}^-, \mathbf{K}^-)}{\mathcal{T}(v, v_{\text{current}})}, \tag{3.14}$$

where $\mathcal{T}(v, v_{\text{current}})$ is the flight time from the current viewpoint v_{current} to a candidate viewpoint v, assuming a constant flight speed of 1 m/s of the UAV. For more technical details on the planning approach, we refer to the work of Popović *et al.* [133]. Note that the reward calculation neither updates the map state nor involves the posterior mean value calculation; therefore, it does not require true measurements to be available at candidate viewpoints. At each planning step, we forward-simulate the map updates for all candidate viewpoints

and select the one with the highest reward as the next viewpoint. The true measurements are then taken at the selected viewpoint to update the map accordingly. We iterate this process until the mission time budget is exhausted. By adaptively merging uninteresting grid cells during mapping to reduce the number of grid cells in the map, our mapping approach allows for accelerated forward-simulation, leading to more efficient view planning.

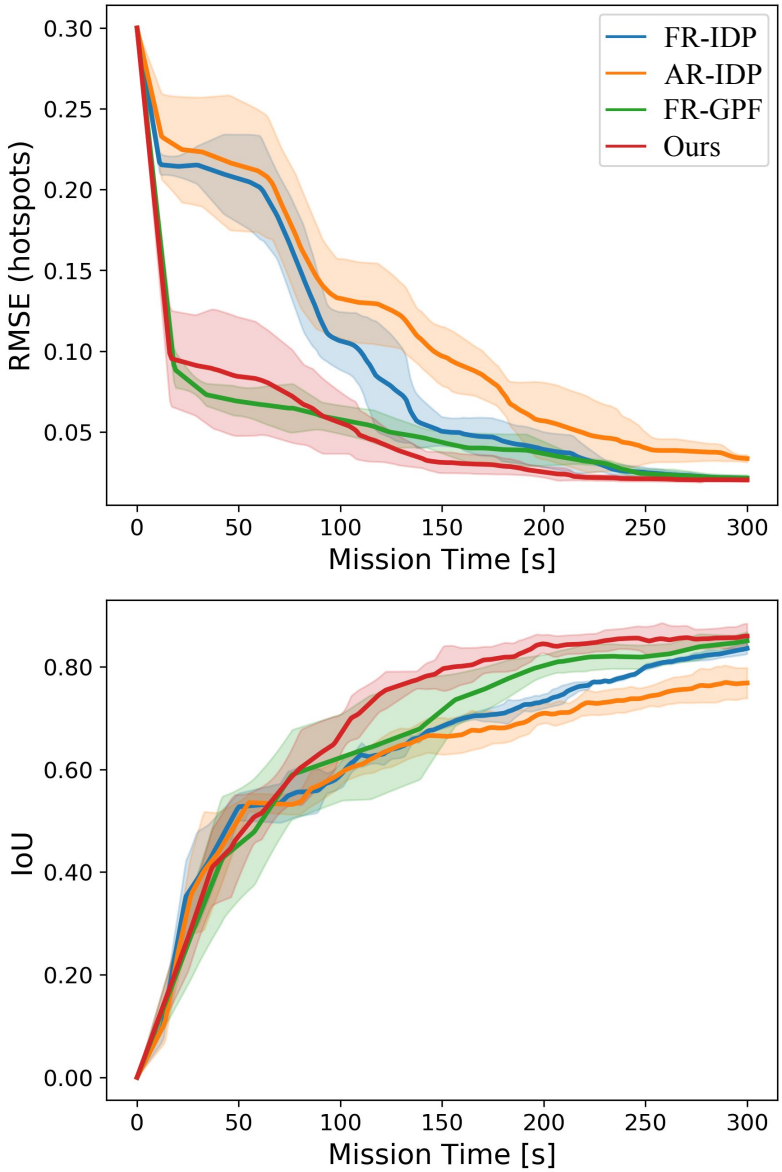


Figure 3.6: Comparison of different mapping approaches used in adaptive view planning. Combining the strengths of both accurate mapping results produced by GP fusion and efficient forward-simulation enabled by our adaptive-resolution approach, our strategy performs the best to efficiently reconstruct hotspot areas in an unknown environment with the highest mapping accuracy (top) and map quality (bottom). Solid lines and shaded regions represent means and standard deviations over 10 trials.

We conduct experiments on 10 simulated Gaussian random fields and plot the evolution of RMSE (hotspots) and IoU over mission time in Figure 3.6. The mission time is defined as the sum of planning time, mapping time, and flight time, reflecting the total operational cost for autonomous UAV-based monitoring. The results show that planning using our mapping approach consistently achieves the best IoU and RMSE (hotspots) scores with the shortest mission time, which is favorable for autonomous monitoring tasks using resource-constrained UAVs. Planning using our approach outperforms *FR-GPF* due to more efficient map updates, which significantly accelerates forward-simulation during adaptive view planning. This efficiency is particularly critical in active perception for robot mapping with GPs, where the cost of uncertainty-driven planning can otherwise dominate the total mission runtime. Our results hence highlight the importance of optimizing the mapping pipeline to accelerate decision-making in closed-loop active perception systems. On the other hand, planning baselines using *FR-IDP* and *AR-IDP* shows poorer performance. This degradation stems directly from the inaccuracy of the underlying maps, which are generated using methods that assume independence between grid cells. As already observed in Section 3.2.1, mapping approaches using the independence assumption neglect important spatial correlations and are thus more susceptible to sensor noise. Due to inaccurate mapping, the false positive interesting areas mislead the UAV into a close inspection of actually uninteresting regions. This inaccuracy deprives *FR-IDP* and *AR-IDP* of their advantage in fast planning.

3.3 Related Work

A large body of literature has studied mapping methods for monitoring spatially correlated variables in different application domains [54, 55, 134, 168, 183, 185]. This chapter focuses on online mapping methods suitable for environmental monitoring scenarios. Our new approach introduces a GP fusion method using GP models with integral kernels as priors for adaptive-resolution mapping. The following subsections review previous studies related to these topics.

3.3.1 Gaussian Processes Mapping

Grid maps are the most commonly used map representation for robot mapping [109]. Despite their successful applications, traditional occupancy grid models assume the stochastic independence of grid cells, mainly to enhance computational efficiency [117]. This representation does not capture the spatial correlations commonly found in natural phenomena, e.g., distributions of temperature, weed density, or humidity. To address this, GP models can be applied for en-

vironmental monitoring. For instance, GPs are used to incorporate uncertainty and represent spatially correlated measurements for aquatic monitoring [46, 54]. Vasudevan *et al.* [183] apply GP regression to predict elevation on a field where sensory information is incomplete. Other applications include gas distribution mapping [168], occupancy mapping [117], terrain elevation [183], and underwater pipe thickness mapping [185]. Our approach follows these lines by using GPs to model the latent scalar field.

The main limitation of applying standard GP regression for online robot mapping is its cubic computational complexity in the number of measurements [140]. Previous work has tackled this problem by storing measurements in a K-d tree structure and using local models to approximate GPs [117, 155, 183]. To predict the mean and variance of query points, one may only consider nearby measurements, thereby reducing the computational costs. However, local GPs often require performing regression for each query point individually, limiting their parallelization capability. To alleviate this problem, Kim *et al.* [79] propose the concept of extended blocks, which applies GPs to the query points in individual blocks of the map only using the measurements in neighboring blocks. This approach decomposes a large GP into sub-models and applies regression to infer the posterior of each block in parallel. The multiple regression results are then fused using a Bayesian committee machine [181], whose computational complexity scales cubically with the number of query points. Based on the Bayesian committee machine, Wang *et al.* [192] introduce test-data octrees to prune nodes of the same state to condense the number of query points in regression, further reducing the redundancy during inference.

In the context of integral kernels, O'Callaghan *et al.* [116] propose GP-based occupancy grid mapping with range sensors that utilizes an integral kernel to process entire beam lines, rather than discretizing them into individual point measurements. This reduces the number of measurements needed for regression and improves efficiency. Most similar to our approach, Reid *et al.* [142] leverage an integral kernel to capture spatial correlations between image areas and infer a high-resolution estimate from a low-resolution measurement in a UAV-based setup. However, inference over the map is still performed using standard GP regression, which suffers from poor scalability, especially with dense image data.

In contrast to the above-discussed regression-based methods, our method leverages GP fusion to reduce the computational burden for online mapping. This procedure removes the need to preserve the measurement history and infer the map posterior from scratch each time new measurements arrive [141]. A key difference in our approach with respect to previous fusion-based works [133, 185] is the proposed integral kernel, which bridges the gap between GP fusion and adaptive-resolution mapping.

3.3.2 Adaptive-Resolution Mapping

In practice, many monitoring scenarios exhibit a non-uniform distribution of information in the environment, i.e., some regions are considered more interesting or informative for mapping than others. Therefore, maintaining a map with constant resolution over the whole environment is redundant and costly. A common method to generate compact map representations is by using tree structures. A well-known algorithm in this category is OctoMap [59], which prunes child nodes with the same state, e.g., occupied, to achieve both memory savings and highly precise maps. Funk *et al.* [44] use the octree structure in an online mapping system that adjusts map resolution based on the occupancy state. Similarly, Chen *et al.* [26] apply quadtrees to build adaptive-resolution 2D maps. The Ndtree proposed by Einhorn *et al.* [35] generalizes these approaches by subdividing any d-dimensional volume recursively with N^d children. Rather than compressing a map only in a postprocessing step, we adapt the map resolution online based on incoming measurements, following the ideas of Einhorn *et al.* [35] and Funk *et al.* [44]. Our approach shares the same motivation, as we tailor the map structure to reduce memory consumption and computation time in applications requiring online mapping, such as adaptive view planning [54, 58, 133].

Previous work in adaptive-resolution mapping assumes spatial independence between cells [35, 44, 59], such that no correlation information needs to be maintained. This substantially simplifies adaptive-resolution mapping at the cost of map quality. In our online mapping setup, the covariance must be correctly modified to account for resolution changes, which is challenging in the GP fusion framework. Popović *et al.* [134] introduce an approach for incrementally fusing variable-resolution measurements into a spatially correlated map. However, their method still considers a fixed-resolution map. In contrast, our strategy supports adaptive-resolution mapping while preserving spatial correlations.

3.4 Conclusion

This chapter introduces a novel approach for online 2D scalar field mapping. Since GP models naturally provide uncertainty modeling that can be used for formulating utility functions, we focus on enhancing mapping efficiency, which is crucial for effective view planning in active perception for robot mapping using GPs as the map representation. We present an integral kernel formulation within the GP fusion method, allowing for incremental and continuous modeling of spatial phenomena in an efficient manner. Unlike standard GP regression based on point measurements, which can be computationally prohibitive for large datasets and online missions, our integral kernel allows for GPs operating on 2D areas.

Combined with an Nd-tree data structure, our formulation facilitates the merging of map information in uninteresting areas, conserving computational and memory resources, while preserving spatial correlations in a theoretically sound fashion.

Experimental results indicate that our approach performs competitively in terms of mapping accuracy, memory efficiency, and computational speed, outperforming traditional methods that rely on fixed-resolution grids, independence assumption, or full GP regressions. To validate its generalization ability, we test our mapping approach using real-world surface temperature measurements on an agricultural field. Furthermore, we show that the efficiency and fidelity of our mapping module significantly benefit downstream tasks such as adaptive view planning, by enabling robots to make better-informed decisions about where to acquire new measurements. This leads to more targeted and efficient measurement acquisition in active perception for robot mapping.

While GP-based approaches have demonstrated effectiveness in 2D scalar field mapping, they are inherently limited in representing high-frequency visual information. They, for example, fall short in supporting photorealistic scene reconstruction. This limitation motivates the exploration of alternative learning-based map representations that better capture fine-grained scene texture and geometry. In the subsequent chapters, we tackle this challenge by integrating active perception with radiance field representations to enable photorealistic reconstruction.

Chapter 4

Next Best View Planning Using Uncertainty Estimation in Image-Based Neural Rendering

With the growing demand for high-fidelity scene reconstruction in robotics and virtual reality applications, implicit neural representations, such as neural radiance fields (NeRFs) [107], are drawing significant interest as a powerful alternative to explicit map structures for representing complex scenes. NeRFs model scenes as continuous volumetric functions, with neural networks encoding both geometric and textural information. This enables photorealistic novel view synthesis after being trained on a set of posed 2D RGB images. Their ability to capture fine-grained details without relying on discretized representations makes them particularly appealing for photorealistic reconstruction.

In the context of active perception for robot mapping, emerging works [89, 124, 139, 177, 213] incorporate uncertainty estimation into NeRFs and exploit it to guide NBV planning [132]. These studies follow an active learning [143] paradigm to collect measurements at the most informative, i.e., most uncertain, viewpoints for periodically retraining a NeRF to improve the scene representation with minimal data. While effective in minimizing data requirements, this retraining-based strategy introduces a significant computational burden. NeRFs require dense samples on rays for rendering and many iterations of gradient-based optimization to converge, making frequent retraining prohibitively slow. As a result, these methods are impractical for online robotic applications.

To overcome the inefficiency caused by per-scene optimization requirements of vanilla NeRF models, an alternative line of research focuses on generalizable image-based neural rendering [145, 182, 194, 207]. Unlike traditional NeRFs, which learn a scene-specific global representation by overfitting to a set of training im-

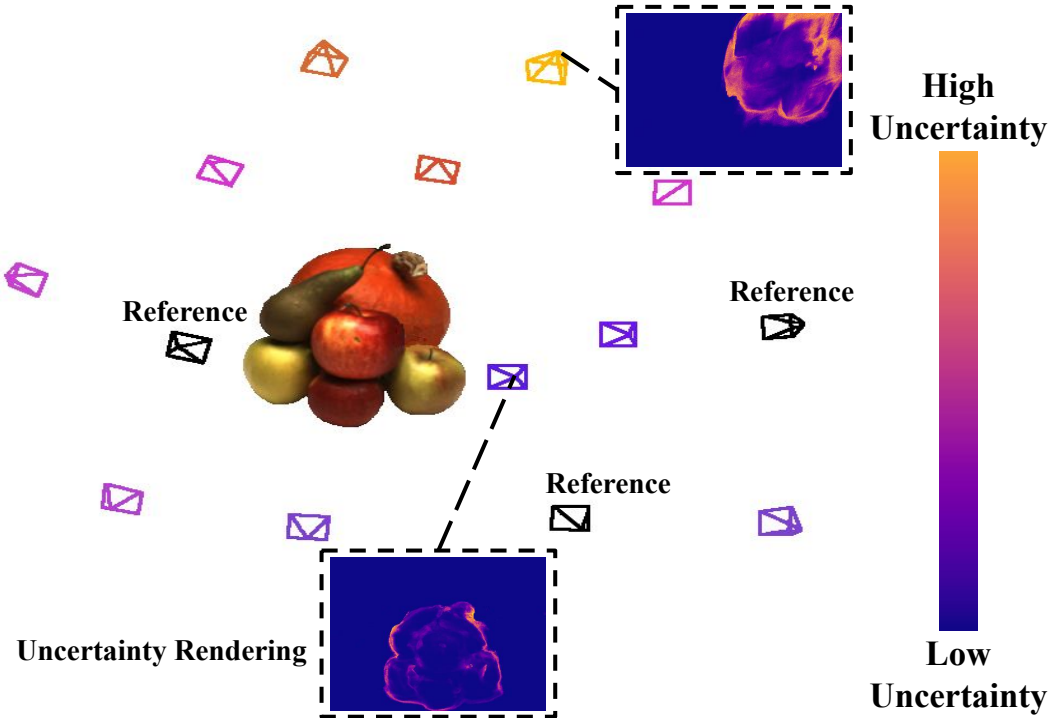


Figure 4.1: Our novel NBV planning approach exploits uncertainty estimation in image-based neural rendering to guide measurement acquisition. Given reference images from the current image collection of the scene (black frustums), our network outputs per-pixel uncertainty estimates at sampled candidate viewpoints (colored frustums). Brighter frustums indicate higher average uncertainty rendered from the viewpoint. Zoom-in boxes illustrate per-pixel uncertainty estimates at the most certain and uncertain viewpoints. By selecting the most informative, i.e., most uncertain, candidate viewpoint at which to take the next measurement, our approach efficiently explores the unknown scene without the need for online map updates.

ages, image-based approaches exploit a shared encoder to map given 2D reference images into latent feature space, upon which the local implicit representation is conditioned. This architecture enables efficient novel view synthesis without the need for per-scene optimization, as the model can render novel views by decoding the latent features extracted from the reference images. By training across a diverse set of scenes, image-based neural rendering models acquire generic scene priors and learn to interpret features in a way that allows them to generalize to previously unseen environments, as introduced in Chapter 2.1.2. Previous work in image-based neural rendering [145, 182, 194, 207] mainly studies improving rendering quality and generalization in offline settings using prerecorded image measurements. However, leveraging the strengths of image-based neural rendering for active perception remains largely unexplored.

The main contribution of this chapter is a novel NBV planning approach bridging the gap between active perception and image-based neural rendering for

robot mapping. A key aspect of our method is a new technique for uncertainty estimation in image-based neural rendering, which enables us to quantify the informativeness of candidate viewpoints without relying on ground-truth images or global scene representations. Intuitively, high uncertainty of color rendering indicates where scene information provided by the closest reference images is insufficient to render the novel view, due to sparse observations, occlusions, or more complex scene details in these areas. As a result, this rendering uncertainty serves as a proxy for identifying unexplored or poorly reconstructed areas of the scene. Therefore, we utilize rendering uncertainty at viewpoints as an informative exploration objective. As shown in Figure 4.1, based on the predicted rendering uncertainty, we actively select the most uncertain candidate viewpoint at each planning step to maximize the information acquired during a measurement acquisition process in an unknown scene.

We make the following three claims:

1. Our uncertainty estimation technique generalizes to unknown scenes, providing an informative proxy for rendering quality of novel views, and is better calibrated compared to baseline approaches.

2. Our uncertainty-guided NBV planning strategy outperforms baseline planning approaches in finding more informative image measurements to represent an unknown scene, given a limited measurement budget.

3. The informative measurements collected using our approach also improve the offline training quality of NeRF models, justifying the effectiveness of our online measurement acquisition strategy.

4.1 Our Approach to View Planning in Image-Based Neural Rendering

We propose a novel NBV planning approach illustrated in Figure 4.2. At each planning iteration, we begin by sampling candidate viewpoints and retrieving their closest reference images from the current image collection. Leveraging the visual information from these references, our image-based neural rendering network predicts per-pixel uncertainty associated with the color prediction for each candidate viewpoint. These uncertainty predictions reflect the network's confidence in rendering novel views from these candidate viewpoints, allowing us to estimate the informativeness of potential measurements. The NBV planning strategy selects the most uncertain candidate viewpoint corresponding to the next measurement, which we add to the image collection. Together with the image

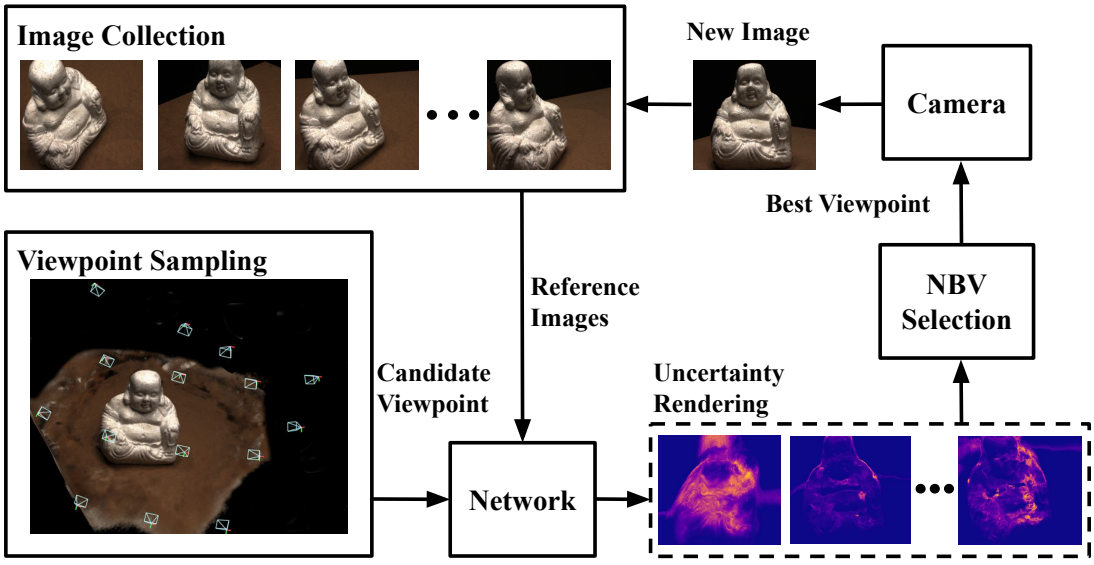


Figure 4.2: Overview of our novel NBV planning approach. We leverage uncertainty estimation in image-based neural rendering to identify areas where the current image collection is insufficient for producing accurate novel view rendering. We use this uncertainty information to actively guide measurement acquisition in unknown scenes. Our image collection and rendering network constitutes the internal map representation in our approach, eliminating the need for map maintenance during online missions.

collection, our neural rendering network retrieves scene information in a purely image-based manner. This enables us to achieve efficient autonomous exploration without maintaining an explicit map or iteratively retraining an implicit neural representation. In the following subsections, we describe our network architecture, training procedure for uncertainty estimation, and NBV planning scheme.

4.1.1 Network Architecture

Our rendering network follows the architectural design of PixelNeRF [207]. Specifically, a shared encoder maps input RGB images into a latent feature space, and an MLP interprets features sampled along each rendering ray to predict scene attributes. PixelNeRF uses a volume rendering technique requiring dense sampling along the ray at predefined intervals, which is inefficient and limits its online applicability. Inspired by Rosu *et al.* [145] and Sitzmann *et al.* [157], we adopt a long short-term memory module [57] to adaptively predict the jumping distance to the next sampling point, avoiding dense sampling in empty space, therefore speeding up the inference of our image-based neural rendering. We illustrate the network architecture in Figure 4.3.

Given a novel viewpoint, we query our current image collection to find the N closest reference images $\mathbf{I}_{n\in\{1,2,\dots,N\}}$ to acquire scene information. We use a shared

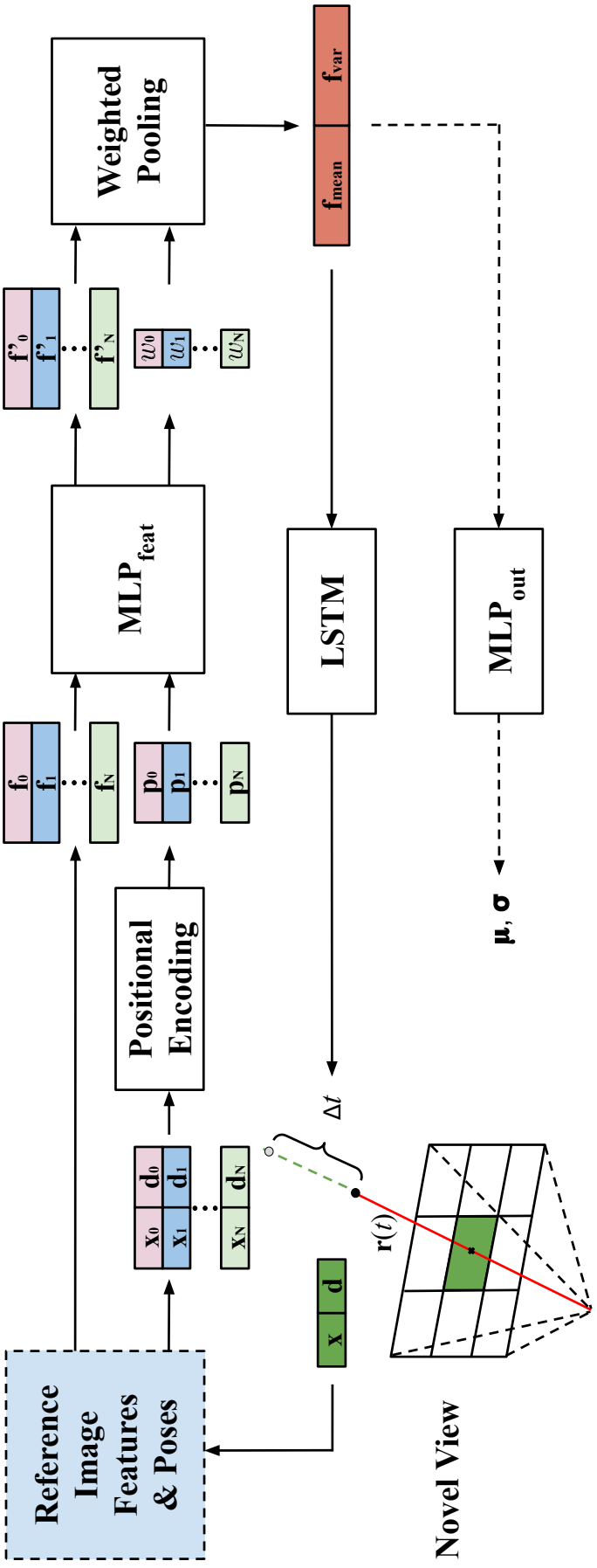


Figure 4.3: Our network architecture. Different colors indicate features from different reference images. Note that the encoder is not explicitly shown. We use a long short-term memory module to predict jumping distance Δt to the next sampling point, given the aggregated feature from all reference images acquired at the current sampling point. After a fixed number of iterations, the aggregated feature at the final point is interpreted to color and uncertainty information. Arrows with dashed lines show the forward pass happening only in the last iteration.

convolution-based encoder to extract latent feature volume $\mathbf{F}_n \in \mathbb{R}^{H\times W\times L}$ from each reference image $\mathbf{I}_n$, where H and W are feature volume's spatial resolution, and L is the channel dimension. We parameterize a ray emitted from the novel viewpoint as $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$, where $\mathbf{o} \in \mathbb{R}^3$ is the camera center position, and t is the distance along normalized view direction $\mathbf{d} \in \mathbb{R}^3$. Starting from the close end of the ray $t = t_s$, we transform the sampling point's position $\mathbf{x} = \mathbf{r}(t)$ and view direction $\mathbf{d}$ into each reference image's coordinate using known relative camera poses to get $\mathbf{x}_n$ and $\mathbf{d}_n$, respectively. To better recover high-frequency details of the scene, the point position $\mathbf{x}_n$ is mapped into higher-dimensional space by the positional encoding operation $\boldsymbol{\gamma}(\mathbf{x}_n)$ proposed by Mildenhall *et al.* [107]. By combining it with its view direction, we compose the pose feature $\mathbf{p}_n = (\boldsymbol{\gamma}(\mathbf{x}_n), \mathbf{d}_n)$ for the sampling point expressed in n^{th} reference image's coordinate. To retrieve the latent image feature from reference images, we project $\mathbf{x}_n$ onto the corresponding reference image plane using known camera intrinsics to get its image coordinate $\boldsymbol{\phi}_{\mathbf{x}_n}$, which we use to query the image feature $\mathbf{f}_n = \text{interp}(\mathbf{F}_n, \boldsymbol{\phi}_{\mathbf{x}_n}) \in \mathbb{R}^L$ by grid sampling with bilinear interpolation [207].

The acquired pose feature $\mathbf{p}_n$ and image feature $\mathbf{f}_n$ from each reference image are processed individually by MLP_{feat}. For aggregating features from all reference images, we use the predicted weight $w_n \in [0\,,1]$ and processed feature $\mathbf{f}'_n$ to calculate the weighted mean $\mathbf{f}_{\text{mean}}$ and variance $\mathbf{f}_{\text{var}}$. This operation downweights the feature from less informative reference images, e.g., due to occlusions or large viewpoint differences. Conditioning on the aggregated feature $(\mathbf{f}_{\text{mean}}, \mathbf{f}_{\text{var}})$, our long short-term memory module adaptively predicts the jumping distance Δt to the next sampling point $\mathbf{x} = \mathbf{r}(t + \Delta t)$, thus mitigating the sampling inefficiency commonly seen in volume rendering [107, 207]. We iterate this process a fixed number of times to let the sampling point approach the surface in the scene and acquire depth prediction. We then use MLP_{out} to interpret the aggregated feature queried at the final sampling point into color and uncertainty information, as detailed in the following subsection.

4.1.2 Uncertainty Estimation in Image-Based Neural Rendering

Our uncertainty estimation quantifies the uncertainty inherited from the input data, due to the varying quality of the information provided by the reference images. Specifically, the predicted uncertainty reflects how well the scene content at a target viewpoint is supported by the available reference images. For example, we expect reference images with large viewpoint differences and self-occlusions with respect to the novel viewpoint to lead to blurry rendering results and thus high uncertainty. An illustration of input-dependent uncertainty estimated using

our new approach is shown in Figure 4.4.

The core to model uncertainty in image-based neural rendering is to interpret the RGB prediction as a probabilistic distribution. Given supervision using only posed 2D images, we incorporate input-dependent uncertainty estimation in the image-based neural rendering training process. Considering that the predicted RGB value is normalized between $[0\,,1]$, we model each channel value of the RGB prediction $c_i \in [0\,,1]$, where $i \in \{1,2,3\}$, as an independent logistic normal distribution described by:

$$p(c_i;\mu_i,\sigma_i) = \frac{1}{\sigma_i\sqrt{2\pi}}\,\frac{1}{c_i(1-c_i)}\exp\left(-\frac{(\mathrm{logit}(c_i)-\mu_i)^2}{2\sigma_i^2}\right), \tag{4.1}$$

where $\mathrm{logit}(c_i) = \ln(\frac{c_i}{1-c_i}) \sim \mathcal{N}(\mu_i,\,\sigma_i^2)$ follows a normal distribution, with the mean μ_i and variance σ_i^2 predicted by our network. To train the network, following Kendall *et al.* [77], we minimize the negative log-likelihood $-\log p\,(c_i = y_i \mid \mu_i,\sigma_i)$, given ground-truth RGB channel values $y_i \in [0\,,1]$. For a single pixel RGB prediction, this leads to our photometric loss function formulated as:

$$\mathcal{L} = \sum_{i=1}^{3}\frac{1}{2}\log(\sigma_i^2) + \log\left(y_i\,(1-y_i)\right) + \frac{(\mathrm{logit}(y_i)-\mu_i)^2}{2\sigma_i^2}\,. \tag{4.2}$$

For calculating the loss, the ground-truth RGB channel value is mapped into logit space by $\mathrm{logit}(y_i)$, before which we clamp y_i at $[0.001\,,0.999]$ to ensure numerical stability. By training the rendering network with this loss function using a large amount of image sets, our rendering network learns novel view rendering in a probabilistic manner.

During deployment in unknown scenes, given a novel viewpoint and its reference images, our network predicts mean μ_i and variance σ_i^2, assuming each RGB channel of a pixel is normally distributed in logit space. We sample 100 times from the normal distribution and pass all samples through a sigmoid function to acquire a valid RGB channel value. The mean and variance of the 100 channel values represent our final channel-wise RGB prediction $c_i \in [0\,,1]$, and the corresponding uncertainty estimate $u_i \in [0\,,0.25]$ of the respective pixel. Since the sampling is conducted in the output space, this operation does not notably increase the computational cost of the network's forward pass.

4.1.3 Uncertainty-Guided Next Best View Planning

Our novel NBV planning approach exploits uncertainty estimation in image-based neural rendering to guide efficient and informative measurement acquisition in unknown scenes. Given a limited measurement budget, our uncertainty-guided approach enables the system to prioritize viewpoints that potentially contribute the most to improving scene understanding and representation quality.

For view planning, we consider a hemispherical surface around the scene as our action space. First, our planning procedure initializes the image collection with image measurements at two random viewpoints. For planning the next camera viewpoint, we uniformly sample a fixed number of candidate viewpoints $v_{k\in\{1,2,\dots,K\}}$ within a constrained angular changes around the current camera viewpoint. For each candidate viewpoint, we retrieve up to N closest reference images from our current image collection based on pose proximity. Given the novel viewpoint and corresponding reference images, our pretrained network renders per-pixel uncertainty estimate $\mathbf{U}_{v_k} \in [0\,,0.25]^{H_r \times W_r \times 3}$ following the approach introduced in Section 4.1.2, where H_r and W_r denote the desired rendering resolution. In this setup, we propose a simple yet effective utility function ψ defined as the average uncertainty values rendered at a candidate viewpoint:

$$\psi(v_k) = \frac{1}{H_r \times W_r \times 3} \operatorname{sum}(\mathbf{U}_{v_k}), \tag{4.3}$$

where sum is the summation operation over the entire uncertainty map. We then select the next best viewpoint v^* with the highest utility value for taking a new measurement and add it to our image collection:

$$v^* = \arg\max_{v_k} \psi(v_k). \tag{4.4}$$

A high uncertainty score indicates that the candidate viewpoint cannot be well-rendered by our network given the current image collection, due to undersampling around the viewpoint, i.e., the closest reference images are far away, or the scene is generally complex when observed from the viewpoint. Therefore, a new measurement at the most uncertain viewpoint potentially yields the highest information value for scene representation using our image-based neural rendering network. We iterate this planning procedure until a given measurement budget is exhausted. Note that our approach is agnostic to sampling strategies and can be easily adapted to other task-specific constraints, viewpoint priors, or robotic kinematics, depending on the application scenarios.

4.2 Experimental Evaluation

Our experimental results support our three claims: (i) we show that our uncertainty estimation in image-based neural rendering is informative to rendering quality and generalizes to new scenes; (ii) we show that our uncertainty-guided NBV planning strategy collects informative image measurements using a publicly available real-world dataset and in a simulated environment. To measure the quality of collected images, we evaluate their influence on image-based neural rendering performance at test viewpoints; and (iii) we show the benefit of using

Figure 4.4: Examples of our input-dependent uncertainty estimation in image-based neural rendering. For rendering the same novel view, we select two sets of reference images. The comparison clearly shows how rendering quality depends on the scene information provided by reference images. Reference images with low information value lead to blurry rendering and correspondingly high uncertainty prediction (yellower) from our network. The error map shows the mean squared error between ground truth and rendered RGB (whiter areas indicate higher errors). Our uncertainty prediction is strongly correlated with this error, thus serving as a good proxy for view planning.

Scene No.		8	21	30	31	34	38	40	41	45	55	63	82	103	110	114
SRCC ↑	Entropy	0.16	0.52	0.37	0.29	0.21	0.60	0.39	0.52	0.17	0.47	0.53	0.32	0.42	0.33	0.60
	Confidence	0.83	0.83	0.90	0.80	0.66	0.76	0.81	0.80	0.83	0.78	0.82	0.88	0.48	0.53	0.79
	Ours	**0.84**	**0.89**	**0.93**	**0.88**	**0.86**	**0.87**	**0.83**	**0.86**	**0.89**	**0.91**	**0.91**	**0.93**	**0.73**	**0.83**	**0.89**
AUSE ↓	Entropy	0.50	0.48	0.34	0.42	0.55	0.48	0.50	0.51	0.51	0.41	0.38	0.34	0.47	0.36	0.45
	Confidence	0.25	0.26	0.14	0.18	0.21	0.28	0.27	0.22	0.19	0.23	0.14	0.16	0.23	0.20	0.16
	Ours	**0.17**	**0.18**	**0.05**	**0.11**	**0.12**	**0.19**	**0.14**	**0.13**	**0.11**	**0.15**	**0.08**	**0.08**	**0.18**	**0.12**	**0.11**

Table 4.1: Evaluation of uncertainty estimation strategies across 15 test scenes from the DTU dataset. Best results in bold.

our online collected images to train NeRFs offline. Experimental results indicate that images collected using our planning approach lead to more accurate radiance field reconstruction when compared against baselines.

4.2.1 Training Procedure

We train our network separately on two datasets for the corresponding planning experiments. We first use real-world images with a resolution of 400×300 pixels from the DTU dataset [68]. We follow the data split proposed by PixelNeRF [207] with 88 training scenes and 15 test scenes, in which no shared or similar scenes exist. For each scene, 49 images are collected following a fixed path pattern on a section of a hemispherical surface. We also record our own synthetic dataset, considering 50 ShapeNet [18] models from 4 representative categories: car, motorcycle, camera, and ship. For each model, we record 100 images with a resolution of 200×200 pixels from viewpoints uniformly distributed on the hemispherical action space covering the scene.

We use the Adam optimizer with a learning rate of 10^{-5} and exponential decay of 0.999. For the long short-term memory module, the iteration number during a forward pass is set to 16. The network is implemented in PyTorch and trained with a single NVIDIA RTX A5000 GPU for around 2 days until convergence. Rendering a novel view with the same resolution as the two dataset images takes 0.6 s and 0.3 s, respectively, which is 60 times faster than PixelNeRF [207]. For both training processes, we randomly select 3, 4, or 5 reference images for novel view rendering in the scene. Our network design is agnostic to the number of input reference images; however, we limit the number of reference images to $N = 5$ to restrict the memory consumption during training.

4.2.2 Evaluation of Uncertainty Estimation

Our first experiment is designed to show that our uncertainty estimation strongly correlates with actual rendering error in image-based neural rendering in unknown scenes. This evaluation is crucial for validating uncertainty as a reliable proxy for guiding active perception. To evaluate the quality of uncertainty prediction, we employ two complementary evaluation metrics. First, we use Spearman's rank correlation coefficient (SRCC) [162] to assess the monotonic relationship between the averaged uncertainty estimates and the rendering errors over a test view. A high SRCC indicates that the uncertainty estimates are informative in ranking the viewpoints by their expected rendering quality. As SRCC only captures the global informativeness of averaged uncertainty prediction, the quality with respect to the structural similarity between the per-pixel uncertainty estimate and error is not considered. To evaluate the structural similarity, we report the

area under sparsification error (AUSE) curve [63]. This metric quantifies the pixel-wise agreement between uncertainty and error by progressively removing the most uncertain pixels and observing the impact on average rendering error. A lower AUSE value indicates a stronger correspondence between uncertain and erroneous pixels, implying that the uncertainty and error maps are spatially well-aligned with each other.

For every test scene in the DTU dataset, we generate 100 randomized test sets. Each test set consists of four images randomly selected from the scene, from which we use three as reference images and the remaining one as the test view. For each test view, we compute the mean square error between the rendered and ground-truth image, as well as the average predicted uncertainty across all pixels. We then calculate the SRCC values based on the 100 pairs of average uncertainty and mean square error values obtained from these test sets. SRCC values above 0.8 empirically indicate strong monotonicity, suggesting that higher predicted uncertainty consistently corresponds to higher rendering error. In addition to SRCC, we also report the average AUSE across the 100 test views for each scene to assess the structural fidelity of the uncertainty estimates.

We compare our approach against two alternative uncertainty estimation methods that can be incorporated into image-based neural rendering pipelines. Lee *et al.* [89] propose calculating the entropy of the density distribution of the samples along each ray as uncertainty quantification in NeRFs. We reimplement this entropy calculation in PixelNeRF, which we denote *Entropy* in the experiments. Rosu *et al.* [145] propose learning to predict RGB rendering confidence in image-based neural rendering by defining the loss as a linear combination of the predicted and the ground-truth images. Similar to our approach, this approach learns to assign high confidence values to pixels that are well-supported by the reference images in the rendering process, while low confidence values otherwise. As their network can only handle a fixed number of reference images with small viewpoint changes, we adapt it by replacing our loss function Equation (4.2) with their confidence loss and train the network under the same conditions as introduced in Section 4.2.1. We denote this method as *Confidence*.

Table 4.1 summarizes the results of uncertainty evaluation. Our uncertainty prediction is more informative and better aligned with rendering error compared to the other two methods. The poor performance of the *Entropy* approach is likely due to the fact that the entropy of the density distribution mainly captures uncertainty over scene geometry, while ignoring the uncertainty in RGB modeling. As proven in prior work [180], neural rendering systems can often reconstruct plausible color information even in the presence of inaccurate depth estimates. Consequently, naively incorporating *Entropy* as uncertainty estimation in image-based neural rendering fails to provide useful information about

rendering quality. The superior performance of our approach compared to *Confidence* indicates that our probabilistic interpretation of RGB prediction leads to more consistent uncertainty estimates. We exemplify a qualitative illustration of our uncertainty prediction results in Figure 4.4.

4.2.3 Comparison of Next Best View Planning Strategies

We show that our uncertainty-guided NBV planning collects the most informative images to better represent an unknown scene. For evaluating planning performance, we use collected images and our image-based neural rendering network to render test views. The rendering quality is measured by the peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) [107]. Note that, since the image-based neural rendering network is fixed for test view rendering in all experiments, performance differences arise purely as the consequence of different NBV planning strategies. We compare our uncertainty-guided approach against two non-adaptive heuristic baselines:

- *Ours*: selects the most uncertain candidate viewpoint via our uncertainty prediction as illustrated in Figure 4.1;
- *Max. View Distance*: selects the candidate viewpoint that maximizes the viewpoint distance with respect to all previously visited viewpoints, reducing the redundant information in the image collection;
- *Random*: selects a candidate viewpoint uniformly at random.

We conduct experiments on the DTU dataset and in our simulator with corresponding pretrained networks, respectively. For all planning experiments, we initialize the image collection with two images collected from randomly selected viewpoints and use different planning approaches to take the next image measurements until a given maximum of measurements is reached.

For experiments on the DTU dataset, we set the measurement budget to 9 images, including the 2 images for initializing the image collection. As the DTU dataset provides a limited number of views per scene, we treat all unselected viewpoints as candidates. We apply three different planning strategies to select the next viewpoint from the pool of candidates and add the corresponding view to our image collection. After each viewpoint selection step, we use the current image collection to render at all viewpoints for performance evaluation. We calculate the average PSNR and SSIM with standard deviations. We repeat the experiment 10 times for all 15 test scenes and report the results in Figure 4.5. As shown, NBV planning guided by our uncertainty estimation selects the most informative candidate viewpoint in each step, reflected by better image-based neural rendering quality.

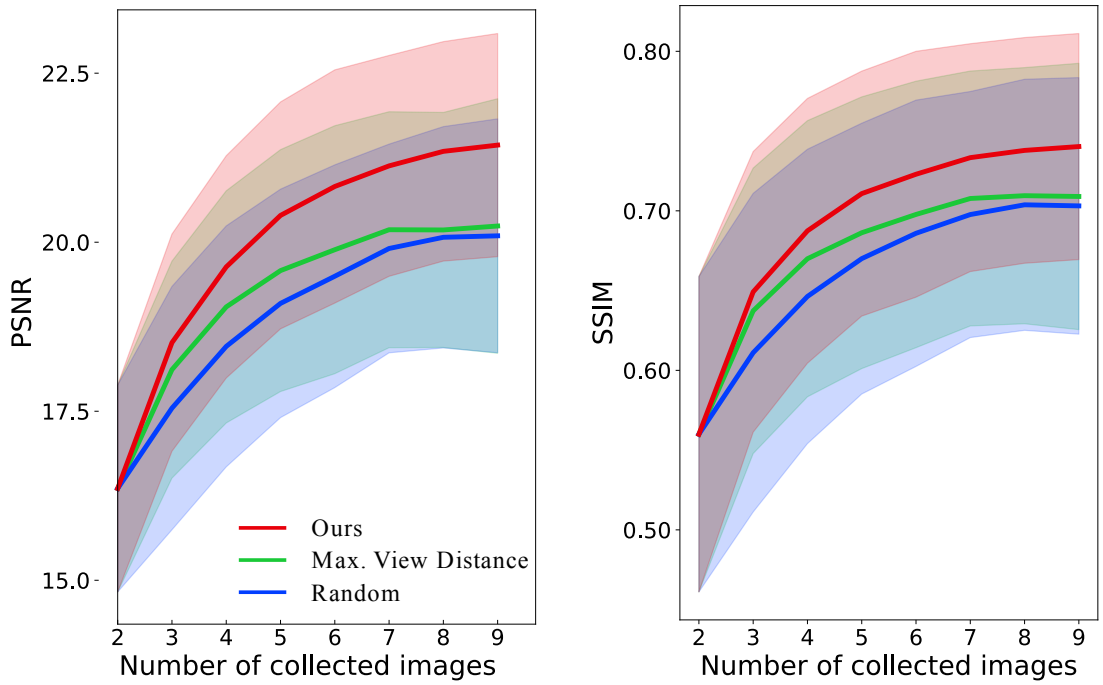


Figure 4.5: Comparison of NBV planners on the DTU dataset. For each test scene, we use our image-based neural rendering network and collected images to render at unselected viewpoints. To evaluate planning performance, we report the average PSNR and SSIM with standard deviations over all test scenes and runs. Note that the large standard deviations are due to the varying rendering difficulty of each scene. Our uncertainty-guided approach finds informative images in the scene, improving scene representations via image-based neural rendering.

To further demonstrate the advantages of our NBV planning approach in a more realistic robotic application scenario, we show the planning experiment in a simulation environment with a continuous action space. We import two different ShapeNet 3D models into the simulator. First, we consider a car model, which belongs to the training category but is not seen during training. Second, to show the generalization ability of our approach, we test our planning approach on an indoor model consisting of a sofa and table. Note that the sofa and table are not in our training data categories. We configure our action space as a hemispherical surface covering the scene and set the measurement budget to 20 images including 2 initialization images. At each planning step, we uniformly sample 50 candidate viewpoints within the interval of maximum 60° angular change with respect to the current camera viewpoint. The three planners select the next viewpoint among the sampled candidates. For our approach, we predict per-pixel uncertainty at 60×60 pixel resolution for each candidate viewpoint using a maximum of 5 closest reference images. One planning step takes 1.5 s in this setting. To evaluate the quality of collected images during online missions, we fix 100 random test views of

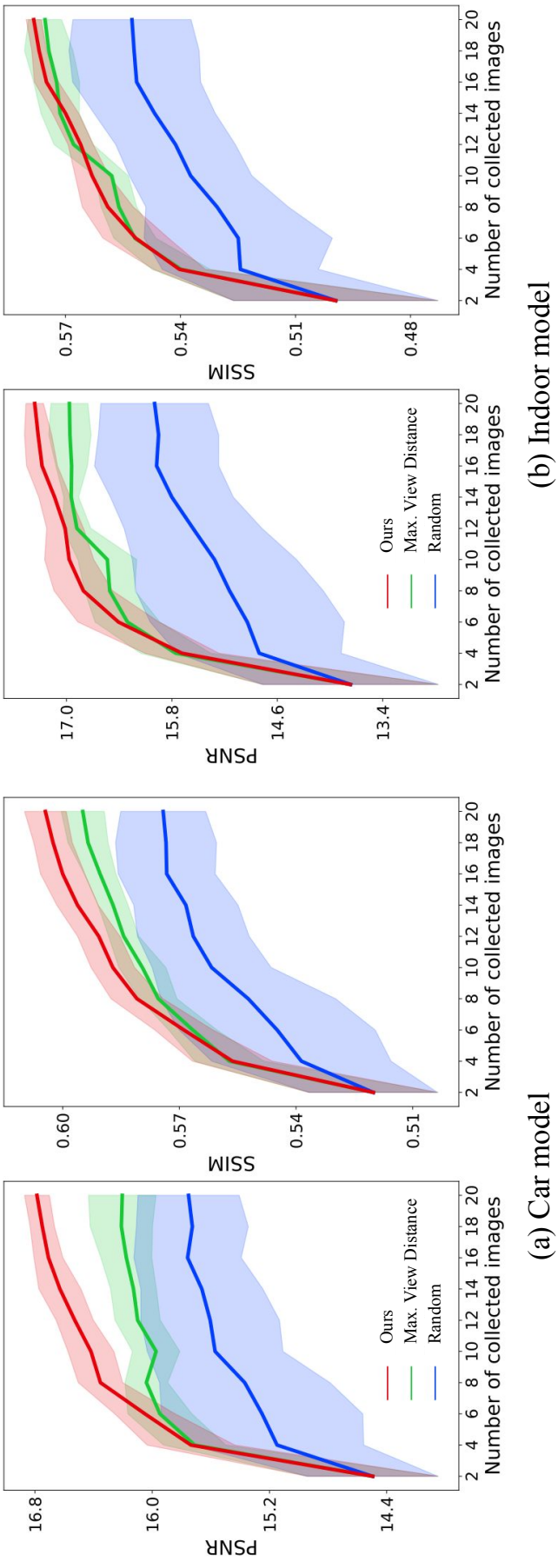


Figure 4.6: Comparison of NBV planners in a ShapeNet-based simulation environment. We conduct the experiments on (a) car and (b) indoor models, respectively. We precollect 100 test views for evaluation purposes. For each test viewpoint, we query a maximum of 5 closest reference images from currently collected images and use our image-based neural rendering network to render the test view. We report the average PSNR and SSIM with standard deviations over all test views and experiment runs. Our uncertainty-guided NBV planning outperforms non-adaptive heuristic baselines in finding more informative images, resulting in higher rendering quality given a limited measurement budget.

the scene. After every 2 measurements, we use our network to render all test views given a maximum of 5 closest reference images from the current image collection and report average PSNR and SSIM with standard deviations to evaluate implicit scene reconstruction quality. We repeat each planning experiment 10 times on the two models, respectively. Figure 4.6 summarizes the planning results in the simulator experiments. Our findings confirm that images collected using our uncertainty-guided approach lead to better image-based neural rendering quality in both scenes. Non-adaptive heuristic approaches cannot efficiently utilize the measurement budget, thus limiting their view planning performance. In contrast, our uncertainty-guided approach collects informative images in a targeted way, resulting in higher test view rendering quality.

4.2.4 Measurement Acquisition for Offline Modeling

In this experiment, we further show that the images collected by our approach improve NeRFs training using limited measurements. Note that, different from uncertainty-guided NBV planning based on NeRFs [89, 124, 139, 177], our uncertainty estimation generalizes to unknown scenes; thus, the measurement acquisition process and NeRF training can be decoupled in our approach. This avoids computationally expensive network retraining during online missions.

After online NBV planning experiments in our simulator, described in Section 4.2.3, we use Instant-NGP [110] to train NeRFs using images collected by the three planning approaches, respectively, under the same training conditions. To evaluate the training results, we render 100 test views using the trained NeRFs. We report the rendering metrics averaged over all experiment runs in Table 4.2 and show examples of rendering results at complex views from the scene in Figure 4.7. Both quantitative and qualitative results verify that our planning strategy for collecting informative images boosts NeRF's performance with limited training data. This indicates the benefits of using our approach to efficiently explore an unknown scene and collect informative images online. The 3D modeling of the scene can be done by training NeRFs offline, after a robotic mission, when computational resources are less constrained.

4.3 Related Work

Our approach bridges the gap between active perception and image-based neural rendering for efficient measurement acquisition in unknown scenes. In this section, we review related work in active perception for robot mapping with a focus on the NBV planning. We discuss current developments in implicit neural representations and the uncertainty estimation in these map representations.

Figure 4.7: Qualitative rendering results of offline trained NeRFs using image collections from three different NBV planners. We select a challenging view from each scene to show the rendering quality differences. Our uncertainty-guided planner adapts the acquisition of new measurements according to the scene structure, in contrast to the heuristic strategies. For example, our approach takes more images of the car bonnet, which has more structural details compared to the car sides and is thus difficult to model using sparse reference images.

		Car	Indoor
PSNR ↑	Max. View Distance	27.37 ± 0.65	30.02 ± 0.55
	Random	25.73 ± 0.83	28.46 ± 0.92
	Ours	$\mathbf{28.35 \pm 0.53}$	$\mathbf{30.46 \pm 0.24}$
SSIM ↑	Max. View Distance	0.925 ± 0.004	0.937 ± 0.003
	Random	0.908 ± 0.012	0.920 ± 0.007
	Ours	$\mathbf{0.934 \pm 0.004}$	$\mathbf{0.941 \pm 0.003}$

Table 4.2: NeRF training results using images collected from our planning experiments in the simulator. Best results in bold.

4.3.1 Next Best View Planning

View planning in active perception for robot mapping is an area of active research [4]. In initially unknown scenes, a common approach is to iteratively select the NBV from a set of candidate viewpoints using a utility function capturing their expected utility based on the current map state.

Isler *et al.* [64] build a probabilistic volumetric map and select the NBV by calculating the utility composed of visibility and the likelihood of seeing new parts of an object from a candidate viewpoint. Similarly, Zaenker *et al.* [208] maintain a voxel map enriched with information of regions of interest. To balance detailed inspection of detected regions of interest with unknown space exploration, they generate candidate viewpoints through targeted sampling around regions of interest in the current map and frontier-based sampling for exploration. To avoid short-sighted decisions, Bircher *et al.* [11] find the NBV in a receding-horizon fashion by generating a random tree of candidate viewpoints, and selecting the branch maximizing the exploration of the amount of unmapped space in a volumetric map. Instead of relying on volumetric map representations, Zeng *et al.* [210] propose a point cloud-based deep neural network to directly predict the utility of candidate viewpoints from the current raw point cloud of the scene. Song *et al.* [161] evaluate the completeness of reconstructed surfaces and extract low-confidence surfaces to guide NBV planning.

All these approaches require explicit, discretized 3D map representations to maintain current information about the scene, which limits their scalability and representation ability. In contrast, our approach utilizes a compact implicit neural representation, conditioned solely on 2D image inputs, for NBV planning to acquire informative measurements.

4.3.2 Implicit Neural Representations

Implicit neural representations parameterize a continuous differentiable signal with a neural network [180]. For example, NeRFs [107] learn a density and radiance field supervised only by 2D images. To render a novel view, NeRFs sample points densely along a camera ray, then predict radiance and density from the position and view direction of each point. The final RGB and depth estimate of the ray is calculated by differentiable volume rendering. As the scene information is encoded in the network parameters, NeRFs overfit to a single scene and require significant training time.

Instead of memorizing a specific scene, image-based neural rendering, e.g., PixelNeRF [207], leverages an encoder to map nearby reference images into latent feature space. After aggregating features from reference images, an MLP is trained to interpret the aggregated features into texture and geometry information at a novel viewpoint. By training across different scenes, image-based approaches generalize well to new scenes without test-time optimization. We exploit the generalization ability of image-based neural rendering to achieve online NBV planning for efficient measurement acquisition in an unknown scene.

4.3.3 Uncertainty Estimation in Neural Representations

Estimating uncertainty in learning-based computer vision tasks is a long-standing problem [77]. Several recent works address uncertainty quantification in NeRFs. S-NeRF [154] proposes learning a probability distribution over all possible radiance fields modeling the scene. To this end, it treats radiance and density as stochastic variables and uses variational inference to approximate their posterior distribution after training. W-NeRF [102] directly learns to predict RGB variance as an uncertainty measure in rendering transient objects in the scene. For image-based neural rendering, Rosu *et al.* [145] introduce a loss function to learn confidence estimation in the rendered images. However, they only consider a fixed number of reference images with small viewpoint changes as inputs, which limits the applicability of their approach in robotics. Smith *et al.* [158] leverage occupancy predictions in image-based neural rendering to estimate geometric uncertainty for active perception. Their approach mainly handles single object shape reconstruction and requires known foreground masks.

Emerging works use uncertainty-guided NBV selection to address NeRF training with a constrained measurement budget. Pan *et al.* [124] and Ran *et al.* [139] model the emitted radiance as a Gaussian distribution and learn to predict the variance by minimizing negative log-likelihood during training. These works add the candidate viewpoint with the highest information gain, i.e., the highest uncertainty reduction, to the existing training data. Instead of learning uncertainty

in parallel to radiance and density, Lee *et al.* [89] and Zhan *et al.* [213] propose calculating the entropy of the density prediction along the ray as an uncertainty measure with respect to the scene geometry. The entropy is used to guide measurement acquisition toward less precise parts. Sünderhauf *et al.* [177] exploit the recent development of fast rendering of Instant-NGP [110] to train an ensemble of NeRFs for a single scene, and measure uncertainty using the variance of the ensemble's prediction, which is utilized for NBV selection.

The above-mentioned approaches address uncertainty-guided NBV selection based on NeRFs. Although these approaches show NeRF model refinement with limited input data, deploying such methods in robotic applications is not straightforward. As the scene information is entirely encoded in the network weights, after each planning step, the uncertainty estimation must be reoptimized to account for newly added measurements, which is time- and compute-consuming. In contrast, our approach incorporates uncertainty estimation in image-based neural rendering to actively select informative image measurements, which are incrementally added to our image collection to better condition the image-based neural rendering. This way, we explore an unknown scene without the need to maintain an explicit map representation or retrain an implicit neural representation.

4.4 Conclusion

In this chapter, we present a novel NBV planning approach using image-based neural rendering for online scene modeling in unknown environments. Central to our approach is a new method for estimating uncertainty in image-based neural rendering, which identifies viewpoints with high predictive uncertainty based on the current set of collected images. Leveraging this powerful tool, we exploit the predicted uncertainty to guide our measurement acquisition.

We demonstrate that our uncertainty estimation is informative to the rendering quality of novel views and generalizes to new scenes. This enables our uncertainty-guided NBV planning to efficiently collect informative images in unknown scenes, which leads to better scene representations via image-based neural rendering. Such a setup offers a significant advantage over previous approaches that require time-consuming retraining of implicit neural representations. Our planning experiments, conducted on both real-world datasets and in simulation, prove that our uncertainty-guided NBV planning scheme effectively finds informative viewpoints in an unknown scene. Measurement acquisition using our approach leads to more accurate scene representations via online image-based neural rendering and offline implicit reconstruction using NeRFs.

One limitation of our current approach is the assumption of a collision-free hemispherical action space, which simplifies view planning. To extend the appli-

cability of active perception for robot mapping to more complex environments, we investigate view planning in unconstrained action spaces, as discussed in Chapter 6. Additionally, our current approach does not differentiate between semantically relevant and irrelevant regions. Integrating semantics with uncertainty estimation in NeRFs, and enabling the robot to actively focus on more task-relevant areas in unknown environments, is particularly important for targeted inspection tasks. We further address this problem in the following chapter.

Chapter 5

Semantic-Targeted Active Implicit Reconstruction

In many applications, including search and rescue, robot manipulation, and precision agriculture, the ability to extract accurate information about the geometry and texture of objects of interest, i.e., objects with specific semantic meanings, is crucial for object-level understanding and downstream task execution. A key challenge in such scenarios is planning a viewpoint sequence to get the most informative measurements targeting the objects of interest, given a limited measurement budget, e.g., operation time or total number of measurements to be integrated. While prior work in active perception for robot mapping has demonstrated the ability to generate high-fidelity map representations, it often lacks semantic awareness in its pipelines, which is important for achieving semantic-targeted active reconstruction.

In this chapter, we address the problem of actively reconstructing objects of one or multiple interesting semantic classes in an initially unknown 3D environment using posed RGB-D camera measurements. Given a limited measurement budget, our goal is to obtain accurate 3D representations of the objects of interest by positioning an onboard camera online, i.e., during a mission, as shown in Figure 5.1. Most existing approaches for active reconstruction [53, 64, 89, 113, 122, 124, 177, 203, 213] aim at reconstructing the whole scene, without distinguishing between the observed objects, such as the approach introduced in Chapter 4. Since they do not incorporate semantics within their planning pipelines, these methods cannot directly use semantic information to target specific objects of interest.

Recently, implicit neural representations [106, 128], such as NeRFs [107], are attracting increasing attention as a compact form for dense scene representation. Follow-up works of NeRFs [20, 110, 118, 173] also address the training inefficiency of implicit neural representations by introducing hybrid structures, which learn

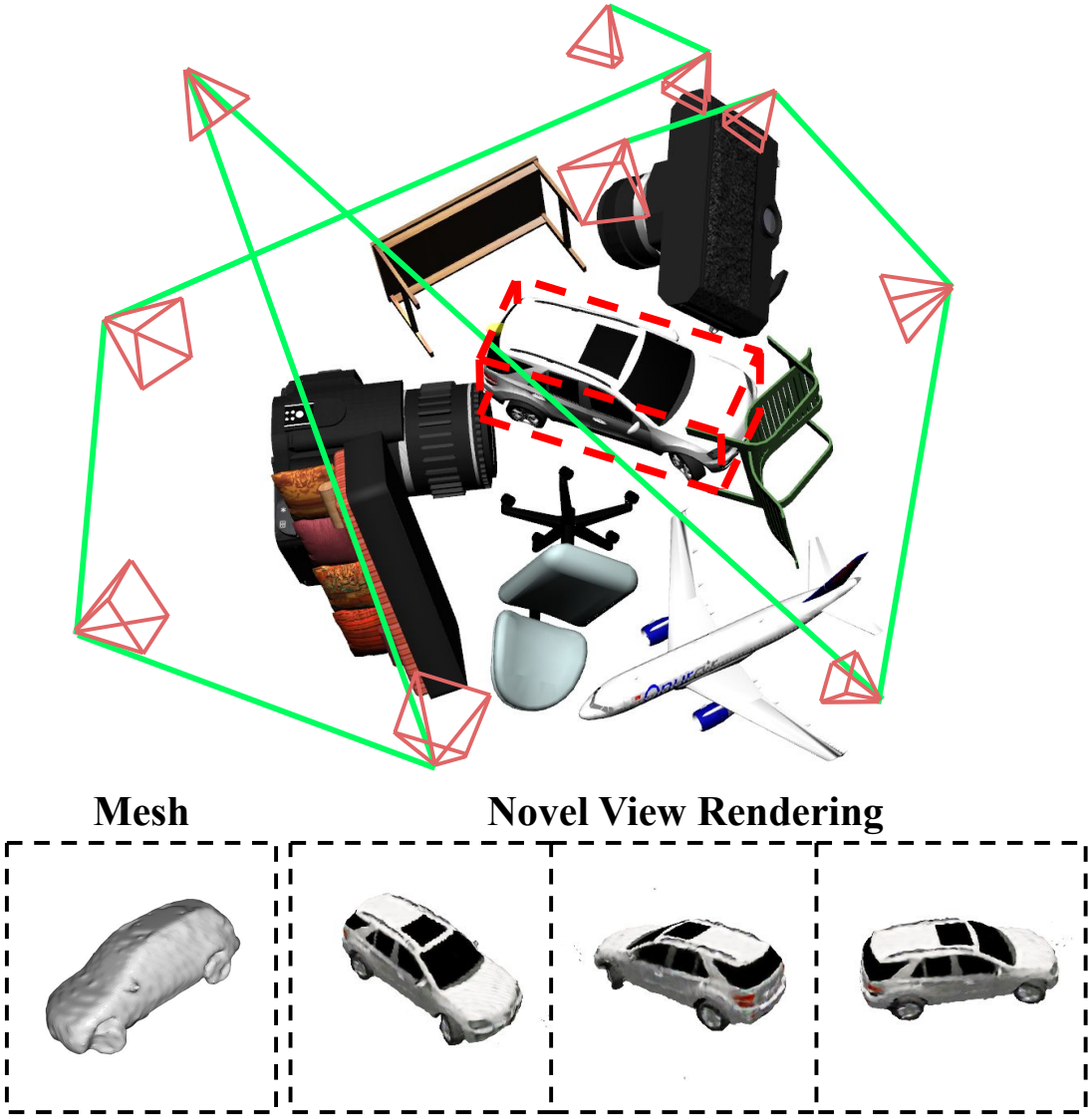


Figure 5.1: Our novel active implicit reconstruction approach targets an object of interest (car) in an unknown environment. We incorporate semantics and uncertainty estimation into our pipeline, enabling view planning to acquire information about the object in a targeted way. The red bounding box identifies the target object. The green line shows the planned path, with pyramids indicating view frustums. With integrated semantics in our implicit neural representation, we can extract mesh and render novel views only for the object of interest, as exemplified in the bottom row.

scene attributes using coarse feature voxel grids combined with shallow MLPs, as mentioned in Chapter 2.1.2. This efficient structure enables deploying implicit neural representations in online robotic tasks [215,218,221], while preserving their continuous representation capabilities. In our approach, we also exploit hybrid implicit neural representations as the map representation for semantic-targeted active implicit reconstruction.

Active implicit reconstruction is an advancing research field [53, 60, 89, 123, 124, 177, 203, 213]. State-of-the-art works adopt NBV planning strategies to find the most informative measurements for training implicit neural representations [53, 89, 124, 177, 203, 213]. While showing promising results, these methods only focus on reconstructing global scenes uniformly. They do not incorporate semantic information, limiting their ability to identify and reconstruct objects of interest in an adaptive and targeted way. In the context of semantics, recent works [8, 156, 189, 217] propose integrating 2D semantic labels into implicit neural representations to enhance semantic understanding capabilities. These approaches show accurate and consistent semantic rendering at novel viewpoints via multi-view learning. However, they have not been used for active reconstruction applications. To bridge the gap between active reconstruction and semantic implicit neural representations, we propose a new approach that enables guiding view planning toward objects of interest in an unknown environment.

The main contribution in this chapter is a novel method called STAIR for semantic-targeted active implicit reconstruction. Given posed RGB-D measurements and corresponding 2D semantic labels, our approach utilizes implicit neural representations to learn occupancy, color, and semantic fields associated with the scene. A key component of our approach is a new utility function for NBV planning using semantic implicit neural representations, which enables trading off between exploring the unknown environment and exploiting information about objects of interest as they are discovered.

We make the following three claims:

1. Our STAIR pipeline shows better performance in terms of reconstructed mesh and RGB rendering quality compared to pure exploration and non-adaptive heuristic baselines that do not consider semantics for view planning.

2. Our method outperforms a state-of-the-art semantic-targeted active reconstruction system using an explicit map representation, both in mapping and planning aspects.

3. Our utility function for planning balances between exploration and exploitation to handle challenging scenes containing many occlusions.

5.1 Our Approach to Semantic-Targeted Active Reconstruction

An overview of our STAIR pipeline is shown in Figure 5.2. Our goal is to actively reconstruct objects of interest in an initially unknown environment using a

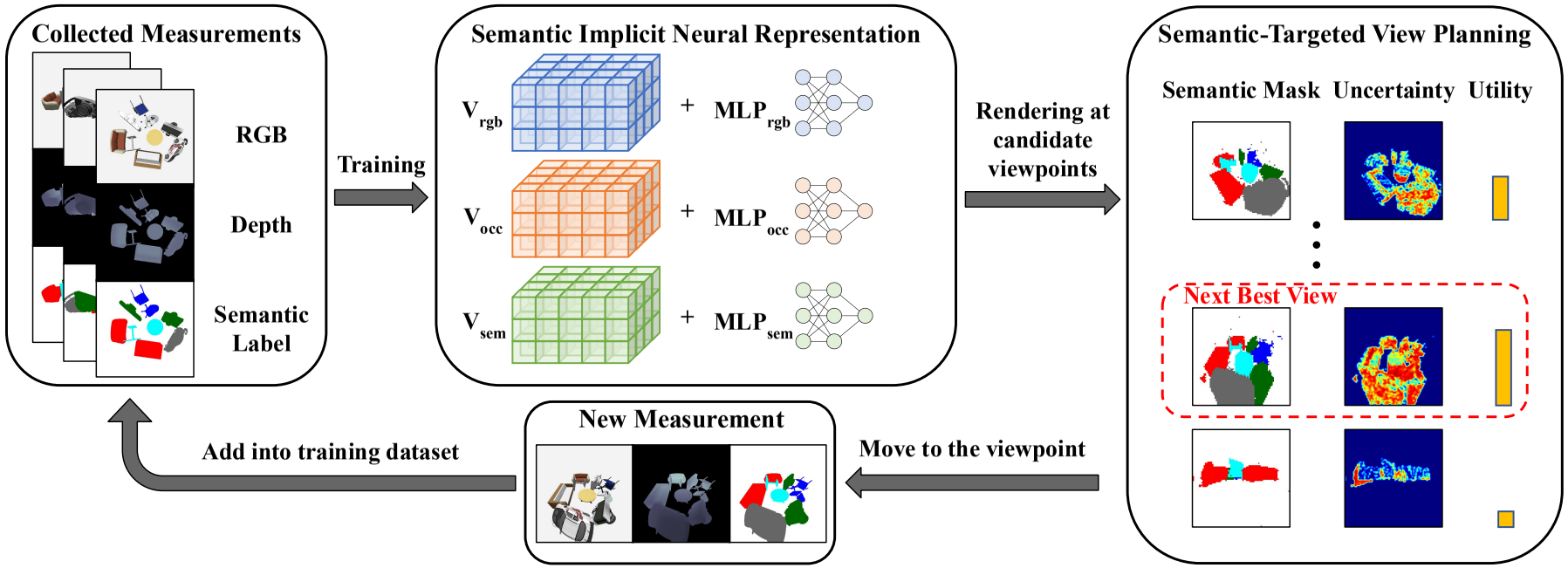


Figure 5.2: Overview of our proposed approach, STAIR. We incrementally train our semantic implicit neural representation using posed RGB-D measurements and their 2D semantic labels. After training, we render semantics and uncertainty at sampled candidate viewpoints. For planning, our utility function considers both overall view uncertainty and the uncertainty from objects of interest. We select the candidate viewpoint with the highest utility value as our next measurement location. We iterate between map representation training and view planning until a maximum allowable number of measurements is reached.

robot equipped with an RGB-D camera. We utilize an implicit neural representation consisting of coarse feature voxel grids and MLPs as our map representation. Given collected posed RGB-D measurements and corresponding semantic labels, we incrementally train our map representation to model the occupancy probability, color, and semantic information in a continuous 3D space. To guide semantic-targeted view planning, we sample candidate viewpoints in a predefined action space and evaluate the utility of each viewpoint based on uncertainty estimates from the occupancy distribution and semantic rendering. The candidate viewpoint with the highest utility value is selected as the location for the next measurement. We iterate between training and planning until a maximum allowable number of measurements is reached.

5.1.1 Semantic Implicit Neural Representation

Similar to DVGO [173], our map representation consists of coarse feature voxel grids and MLPs to balance representation capabilities and training efficiency. We employ voxel grids to preserve local scene features, while the MLPs interpret these features into desired modalities. In our approach, we maintain features for corresponding modalities of the scene: spatial occupancy (occ), RGB color (rgb), and semantics (sem), in three voxel grids $\mathbf{V}_{\text{occ}}$, $\mathbf{V}_{\text{rgb}}$, and $\mathbf{V}_{\text{sem}}$, respectively. For any point in the space, we can query its modality feature by a trilinear interpolation operation interp in the corresponding voxel grid expressed as:

$$\mathbf{f}_m = \text{interp}(\mathbf{x}, \mathbf{V}_m) : (\mathbb{R}^3 \times \mathbb{R}^{T_m \times H \times W \times L}) \rightarrow \mathbb{R}^{T_m}, \tag{5.1}$$

where $m \in \{\text{occ}, \text{rgb}, \text{sem}\}$, $\mathbf{f}_m \in \mathbb{R}^{T_m}$ is the queried modality feature vector at position $\mathbf{x} \in \mathbb{R}^3$, $\mathbf{V}_m$ is the feature voxel grid of corresponding modality with T_m feature channels, and H, W, L are the spatial resolution dimensions.

The queried modality features at point $\mathbf{x}$ are interpreted by modality-specific MLPs into per-point occupancy probability $o(\mathbf{x}) = \text{MLP}_{\text{occ}}(\boldsymbol{\gamma}(\mathbf{x}), \mathbf{f}_{\text{occ}}) \in [0, 1]$, RGB color $\mathbf{c}(\mathbf{x}) = \text{MLP}_{\text{rgb}}(\boldsymbol{\gamma}(\mathbf{x}), \mathbf{f}_{\text{rgb}}) \in [0, 1]^3$, and semantic probability vector $\mathbf{s}(\mathbf{x}) = \text{MLP}_{\text{sem}}(\boldsymbol{\gamma}(\mathbf{x}), \mathbf{f}_{\text{sem}}) \in [0, 1]^P$, with P as the number of total semantic classes. We use a positional encoding function [107] $\boldsymbol{\gamma} : \mathbb{R}^3 \rightarrow \mathbb{R}^{21}$ to map position $\mathbf{x}$ into a higher-dimensional space. Note that we assume Lambertian surfaces and do not consider view-dependent color emission in this approach.

5.1.2 Training of Map Representation

Our map representation is updated online during a mission. Given a set of posed RGB-D measurements obtained by the robot camera and their semantic labels, we jointly train our feature voxel grids and MLPs using differentiable volume rendering [107]. To render color, depth, and semantics for a ray $\mathbf{r}$ cast from a

measurement viewpoint, we uniformly sample N points $\mathbf{x}_{i\in\{1,2,\dots,N\}}$ along the ray with $d(\mathbf{x}_i)$ as the depth value from the sampling point $\mathbf{x}_i$ to its viewpoint origin. Following UNISURF [118], occupancy-based volume rendering for predicted color $\mathbf{C}(\mathbf{r})$, depth $D(\mathbf{r})$, and semantic probability $\mathbf{S}(\mathbf{r})$ observed from ray $\mathbf{r}$ is given by:

$$\mathbf{C}(\mathbf{r}) = \sum_{i=1}^{N} w(\mathbf{x}_i)\, \mathbf{c}(\mathbf{x}_i)\,, \tag{5.2}$$

$$D(\mathbf{r}) = \sum_{i=1}^{N} w(\mathbf{x}_i)\, d(\mathbf{x}_i)\,, \tag{5.3}$$

$$\mathbf{S}(\mathbf{r}) = \sum_{i=1}^{N} w(\mathbf{x}_i)\, \mathbf{s}(\mathbf{x}_i)\,, \tag{5.4}$$

with:

$$w(\mathbf{x}_i) = o(\mathbf{x}_i)\, T(\mathbf{x}_i)\,, \quad T(\mathbf{x}_i) = \prod_{j<i} \left(1 - o\left(\mathbf{x}_j\right)\right)\,, \tag{5.5}$$

where $w(\mathbf{x}_i)$ is the weight of modality value at $\mathbf{x}_i$ and $T(\mathbf{x}_i)$ is accumulated transmittance, indicating the probability of ray reaching $\mathbf{x}_i$ without being blocked by built surfaces.

We supervise the training using the loss terms:

$$\mathcal{L}_{\text{rgb}} = \frac{1}{|\mathcal{R}|} \sum_{\mathbf{r}\in\mathcal{R}} \left\| \mathbf{C}(\mathbf{r}) - \hat{\mathbf{C}}(\mathbf{r}) \right\|_2\,, \tag{5.6}$$

$$\mathcal{L}_{\text{depth}} = \frac{1}{|\mathcal{R}|} \sum_{\mathbf{r}\in\mathcal{R}} \left\| D(\mathbf{r}) - \hat{D}(\mathbf{r}) \right\|_1\,, \tag{5.7}$$

$$\mathcal{L}_{\text{sem}} = \frac{1}{|\mathcal{R}|} \sum_{\mathbf{r}\in\mathcal{R}} \text{CE}\left(\mathbf{S}(\mathbf{r}),\, \hat{\mathbf{S}}(\mathbf{r})\right)\,, \tag{5.8}$$

where $\hat{\mathbf{C}}(\mathbf{r})$, $\hat{D}(\mathbf{r})$, and $\hat{\mathbf{S}}(\mathbf{r})$ are the recorded color, depth, and semantic label respectively of ray $\mathbf{r}$ in the measurements; CE refers to the cross entropy loss [97], and $\mathcal{R}$ denotes the set of rays in the training batch. The total training loss is:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{rgb}} + \lambda_2 \mathcal{L}_{\text{depth}} + \lambda_3 \mathcal{L}_{\text{sem}}\,, \tag{5.9}$$

with the factors $\lambda_1, \lambda_2, \lambda_3$ balancing the weight of each term in the loss function. Note that, although we focus on objects of interest, the reconstruction of other regions is necessary for view planning under occlusions present in the scene.

We incrementally train our map representation for a constant number of iterations when a new measurement arrives. To avoid overfitting to the latest measurement, we collect our training batch $\mathcal{R}$ for each training iteration from both previous measurements and the latest measurement. We assign the probability of sampling each training ray example as being inversely proportional to its total sampled time to ensure uniform sampling across the whole training dataset. After training, our map representation is used for semantic-targeted view planning, as introduced in the next subsection.

5.1.3 Semantic-Targeted View Planning

A key aspect in our approach is a utility function that adaptively guides view planning by trading off between exploration and exploitation. We first introduce our sampling strategy for generating candidate viewpoints and then elaborate on how we calculate utility values for viewpoint selection.

To generate candidate viewpoints, we adopt a two-stage sampling strategy. We first uniformly sample N_{uni} candidate viewpoints on the object-centric hemispherical surface action space. We evaluate the individual utility of each viewpoint and select the viewpoints of the top K utility values. We then resample N_{re} new candidate viewpoints around each of these viewpoints to obtain a fine-grained utility evaluation. Finally, the candidate viewpoint with the highest utility value is selected as the next measurement location.

Our utility quantification considers uncertainty estimates and semantic rendering. Uncertainty estimation indicates parts of the scene that are unexplored or still not well-reconstructed. At the same time, semantic rendering provides masks to distinguish objects of interest from other uninteresting regions, allowing for view planning in a targeted way. We derive the uncertainty estimates from our trained occupancy field. For a candidate viewpoint v_k, we sample N_{pt} points on each of N_{ray} rays cast from the viewpoint. We define the uncertainty at each sampling point $\mathbf{x}_i$ as its entropy:

$$H_{\text{pt}}(\mathbf{x}_i) = -o(\mathbf{x}_i)\ln(o(\mathbf{x}_i)) - \bar{o}(\mathbf{x}_i)\ln(\bar{o}(\mathbf{x}_i))\,, \tag{5.10}$$

where $\bar{o} = 1 - o$ is the complementary occupancy probability. Note that we do not consider the entropy of sampling points behind the built object surface. Thus, the total entropy along a ray $\mathbf{r}$ is:

$$H_{\text{ray}}(\mathbf{r}) = \sum_{i=1}^{N_{\text{pt}}} T(\mathbf{x}_i)\, H_{\text{pt}}(\mathbf{x}_i)\,, \tag{5.11}$$

where T is the accumulated transmittance term introduced in Equation (5.5). The sum of uncertainty rendered at viewpoint v_k is:

$$U_{\text{er}}(v_k) = \sum_{i=1}^{N_{\text{ray}}} H_{\text{ray}}(\mathbf{r}_i)\,, \tag{5.12}$$

which we define as our exploration (er) score. This term does not distinguish between the uncertainty values associated with different objects. To account for objects of interest based on their semantic meaning, we apply a mask to the uncertainty according to whether or not the objects are relevant for semantic-

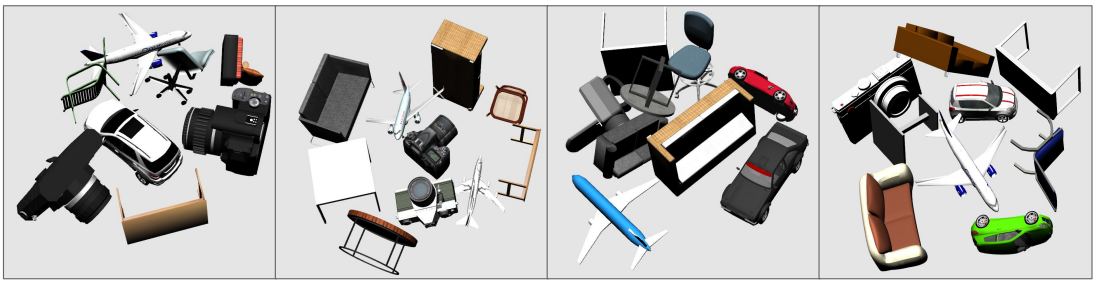


Figure 5.3: Four different scenes used in our main planning experiments. Our interesting semantic classes are: car for Scene 1, camera for Scene 2, sofa for Scene 3, car and airplane for Scene 4.

targeted active planning:

$$U_{\text{et}}(v_k) = \sum_{i=1}^{N_{\text{ray}}} H_{\text{ray}}(\mathbf{r}_i)\delta(\mathbf{r}_i)\,, \tag{5.13}$$

$$\delta(\mathbf{r}_i) = \begin{cases} 1 & \text{if } \operatorname{argmax}\left(\mathbf{S}(\mathbf{r}_i)\right) \in \mathcal{T} \\ 0 & \text{otherwise} \end{cases}, \tag{5.14}$$

where $\mathbf{S}(\mathbf{r}_i)$ is the predicted semantic probability vector obtained using Equation (5.4) and $\mathcal{T} \subseteq \{1, 2, \ldots, P\}$ is a set of identifiers for the interesting semantic classes. We denote the sum of pixel-wise uncertainty from the objects of interest as our exploitation (et) score, which guides view planning toward target objects.

To trade off between exploring the unknown environment and exploiting information about objects of interest as they are discovered, we compute the utility value of a candidate viewpoint as the sum of exploitation and weighted exploration score, with ε as the weight factor:

$$\psi(v_k) = U_{\text{et}}(v_k) + \varepsilon U_{\text{er}}(v_k)\,. \tag{5.15}$$

5.2 Experimental Evaluation

Our experimental results support our three claims: (i) we show the superior performance of our approach in terms of rendering and mesh quality by considering semantic information for view planning; (ii) we show that our approach based on implicit neural representation outperforms approaches using explicit map representations for semantic-targeted reconstruction tasks; and (iii) we validate the effectiveness of our utility formulation that balances between exploration and exploitation in view planning, especially in challenging scenes with occlusions.

5.2.1 Experimental Setup

We spawn ShapeNet [18] models of different semantic classes with random poses to build test scenes. We consider 7 semantic classes in our simulator: car, airplane, sofa, chair, table, camera, and background. Four test scenes used in the planning experiments are shown in Figure 5.3. All scenes consider a bounding box size of $3\,\mathrm{m} \times 3\,\mathrm{m} \times 3\,\mathrm{m}$. We set our camera action space as an object-centric hemispherical surface with 2 m radius and camera viewpoints targeting the scene origin. All RGB-D measurements are captured at 400×400 pixel resolution. To acquire the semantic labels, pretrained semantic segmentation models can be applied; however, in this work, we use ground-truth semantics from the simulator to focus on evaluating planning performance.

We use a grid size of $128 \times 128 \times 128$ for all three feature voxel grids. We set the feature channels as $T_{\mathrm{occ}} = 3$, $T_{\mathrm{rgb}} = 6$, and $T_{\mathrm{sem}} = 7$. The $\mathrm{MLP}_{\mathrm{rgb}}$ comprises two hidden layers with 128 channels, while $\mathrm{MLP}_{\mathrm{occ}}$ consists of two hidden layers with 32 channels. We simply use an identity mapping as $\mathrm{MLP}_{\mathrm{sem}}$ and no positional encoding for modeling semantics since the semantic field is smooth and exists in a low-frequency domain. We set $\lambda_1 = 1.0, \lambda_2 = 0.1$, and $\lambda_3 = 1.0$ in Equation (5.9). For each training iteration, we use a batch size of 8000 with 4000 training examples from all previous measurements and 4000 training examples from the current measurement. We train our map representation for 200 steps before conducting view planning, which takes approximately 5 s and 2 GB video memory with our PyTorch implementation running on an NVIDIA RTX A5000 GPU.

For candidate viewpoint sampling strategy introduced in Section 5.1.3, we set $N_{\mathrm{uni}} = 100$, $K = 10$, and $N_{\mathrm{re}} = 10$, giving a total of 200 viewpoints. To render semantic and uncertainty maps at a candidate viewpoint, we use $N_{\mathrm{ray}} = 80 \times 80$ and $N_{\mathrm{pt}} = 200$. One planning step takes around 2 s under this sampling and rendering configuration. The exploration weight ε in Equation (5.15) is 0.2. We select car in Scene 1, camera in Scene 2, sofa in Scene 3, car and airplane in Scene 4 as the interesting classes for semantic-targeted active reconstruction. The maximum allowable number of planning steps is set to 10 for all experiments.

We evaluate the reconstruction results with test view rendering performance and mesh quality. We report the PSNR [107] as the rendering metric and use the F1-score to measure overall mesh quality. Since our goal is to reconstruct objects of interest, we only consider these objects in the metrics calculations. Hence, when rendering at test viewpoints or extracting meshes from our trained map representation, we only keep objects of interest by setting the occupancy probability of points with uninteresting semantic predictions to zero.

For calculating PSNR, we render color images at 100 uniformly distributed test viewpoints and compare the predictions with ground-truth images. We aver-

age the PSNR over all test views as the final rendering metric. For mesh quality evaluation, we first extract the mesh of objects of interest from our trained occupancy field using multiresolution isosurface extraction [106] with a threshold of 0.5. We uniformly sample 10^6 points on both the extracted mesh and the ground-truth mesh. The precision is calculated as the fraction of points on the extracted mesh that are closer than a threshold distance to points on the ground-truth mesh. Similarly, the completeness is the fraction of points on the ground-truth mesh that match points on the extracted mesh within a threshold distance. We use 1 cm as the threshold value for precision and completeness calculations. Finally, the F1-score is the harmonic mean of precision and completeness.

5.2.2 Comparison of Active Implicit Reconstruction

Our first experiment shows that our semantic-targeted view planning method achieves better reconstruction quality in terms of rendering performance and mesh quality compared to pure exploration and non-adaptive heuristic baselines that do not consider semantics. The map representations and training configurations are the same for all methods; hence, the reconstruction quality differs purely as the consequence of the collected measurements using different planning strategies. We consider the following planning methods:

- *Ours*: selects the viewpoint with the highest utility value defined in Equation (5.15);
- *Exploration*: selects the viewpoint with the highest exploration score as calculated by Equation (5.12), following the strategy proposed by Lee *et al.* [89];
- *Fixed Pattern*: follows the spiral pattern viewpoint sequence to cover the hemispherical action space;
- *Max. View Distance*: selects the viewpoint that maximizes the viewpoint distance to all previously visited viewpoints;
- *Uniform*: selects a random viewpoint from uniformly sampled candidate viewpoints in the action space.

For all experiment runs, we start with a measurement from the top viewpoint and use different planning methods to select the next viewpoint to acquire a new measurement, which, together with all previous measurements, is used to train our map representation. We evaluate reconstruction performance after every planning step. For each test scene and planning method, we run 5 trials and report the average PSNR and F1-score with standard deviations along the planning steps.

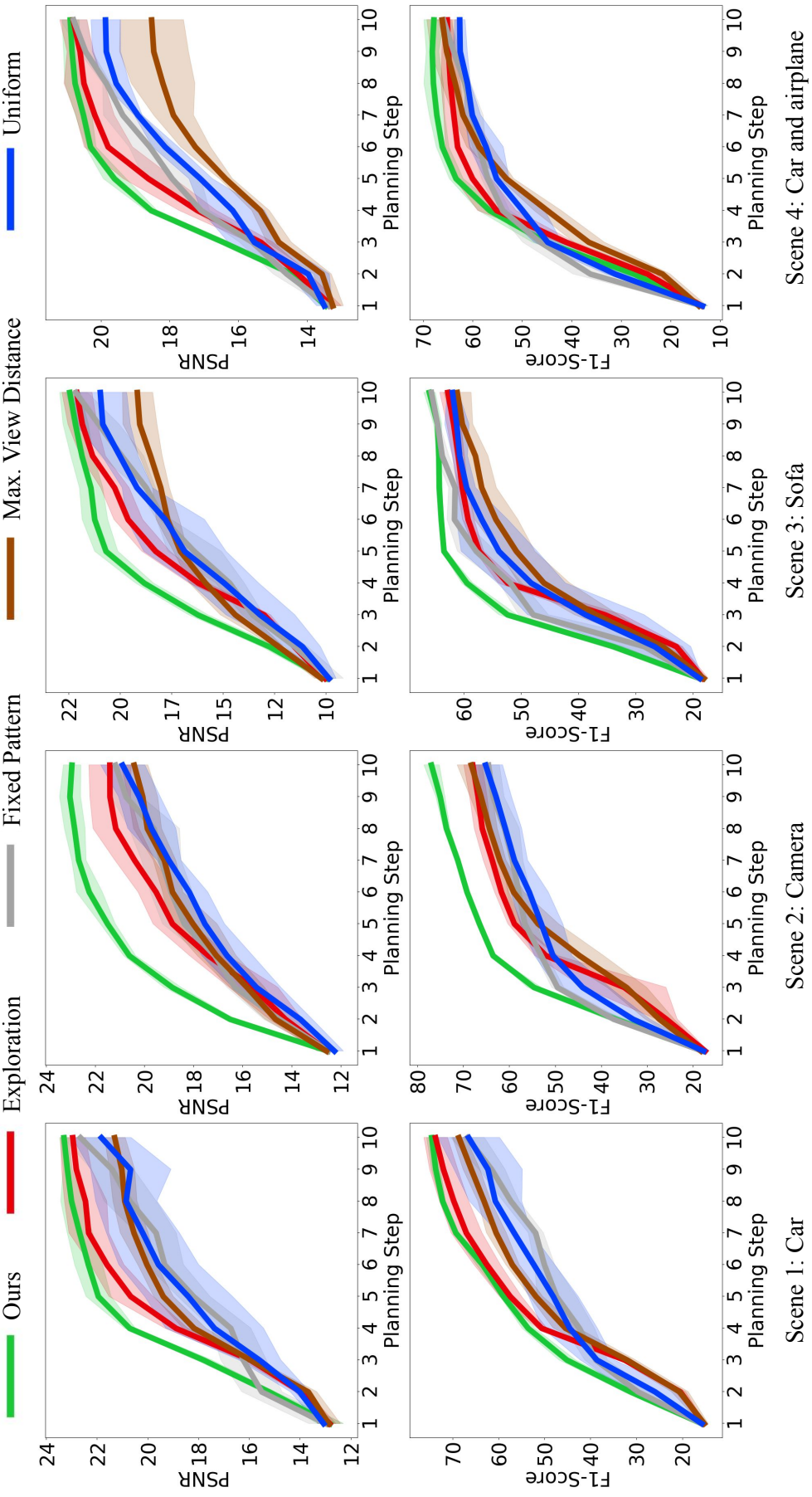
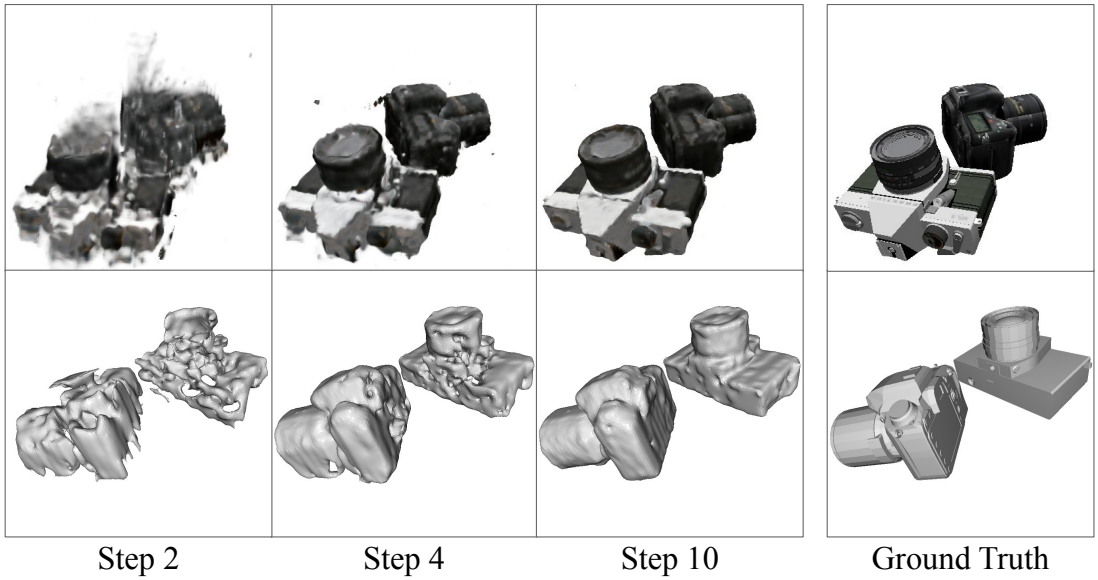


Figure 5.4: Comparison of reconstruction quality of objects of interest using different planning strategies in the four test scenes shown in Figure 5.3. We report the average PNSR and F1-score at each planning step. Solid lines show means over 5 trials and shaded regions indicate standard deviations. Our semantic-targeted approach exploits semantics in our implicit neural representation to achieve targeted view planning, leading to better and more stable reconstruction performance.



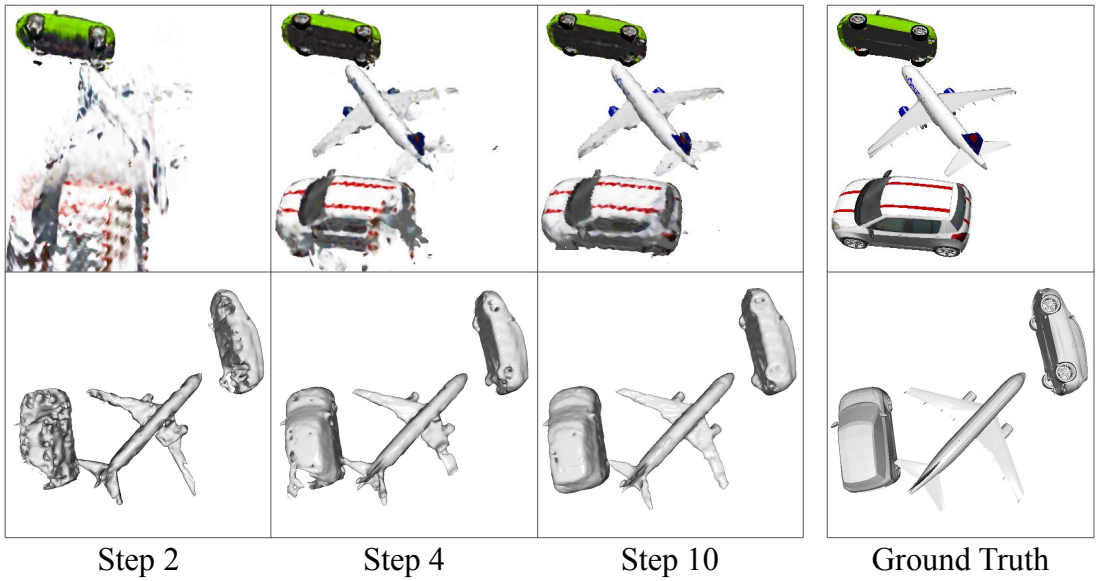


Figure 5.5: Qualitative results of our approach showing how novel view rendering (top) and meshes (bottom) improve along planning steps during a mission. Our approach collects informative measurements about objects of interest in a targeted way to achieve high-quality reconstruction.

The experiment results are given in Figure 5.4. We plot the mapping quality over planning steps in terms of PSNR for rendering and F1-score for mesh evaluation. NBV planning guided by our approach shows steeper-rising metric curves, indicating more efficient reconstruction compared to baselines that do not con-

sider semantic information. This verifies that our STAIR pipeline benefits from integrating semantics in an implicit neural representation to achieve semantic-targeted active reconstruction. Our approach has the lowest standard deviations across all scenes, indicating its robust performance. In Figure 5.5, we show two examples of how novel view rendering and object meshes improve along planning steps using our approach.

5.2.3 Comparison of Semantic-Targeted Explicit Reconstruction

In this experiment, we compare our STAIR with a semantic-targeted active explicit reconstruction approach to show the advantages of using an implicit neural representation for our task. Specifically, we compare against the approach of Zaenker *et al.* [208], which we denote as *STE* to indicate semantic-targeted planning based on explicit map representations. *STE* fuses RGB-D measurements and 2D semantic labels into an explicit semantic occupancy grid map and biases planning toward the objects of interest as they are built in the map by assigning higher utility to unknown voxels close to objects of interest. For comparability, we use the same grid size of $128 \times 128 \times 128$ for their map.

To further investigate the sources of performance difference between our approach and *STE*, we cross-validate these two active reconstruction approaches by combining measurements collected by each approach with the other mapping method. After the online planning experiments, we fuse the measurements collected by our approach into an explicit occupancy map used in the *STE* approach. We denote this combination as *Ours (Explicit)*. The result of this combination can inform us whether the performance gain originates from our view planning results. Similarly, we use the measurements collected by the *STE* approach to train our implicit neural representation, which we denote as *STE (Implicit)*. This combination exposes how different map representations influence the reconstruction performance when the measurement inputs are held constant.

The results are shown in Figure 5.6, where we only compare the F1-score of mesh reconstruction after each planning step, as the explicit map is incapable of photorealistic rendering. Our approach performs better than the *STE* method. The performance gain can be decomposed into two aspects. First, comparing *STE (Implicit)* and *STE* suggests that, given the same measurements, our implicit neural representation improves reconstruction quality compared to explicit occupancy mapping. This justifies the choice of using implicit neural representations in our active reconstruction approach. Second, as seen by comparing *Ours (Explicit)* and *STE*, even when using explicit occupancy mapping, measurements acquired using our planning approach lead to better reconstruction

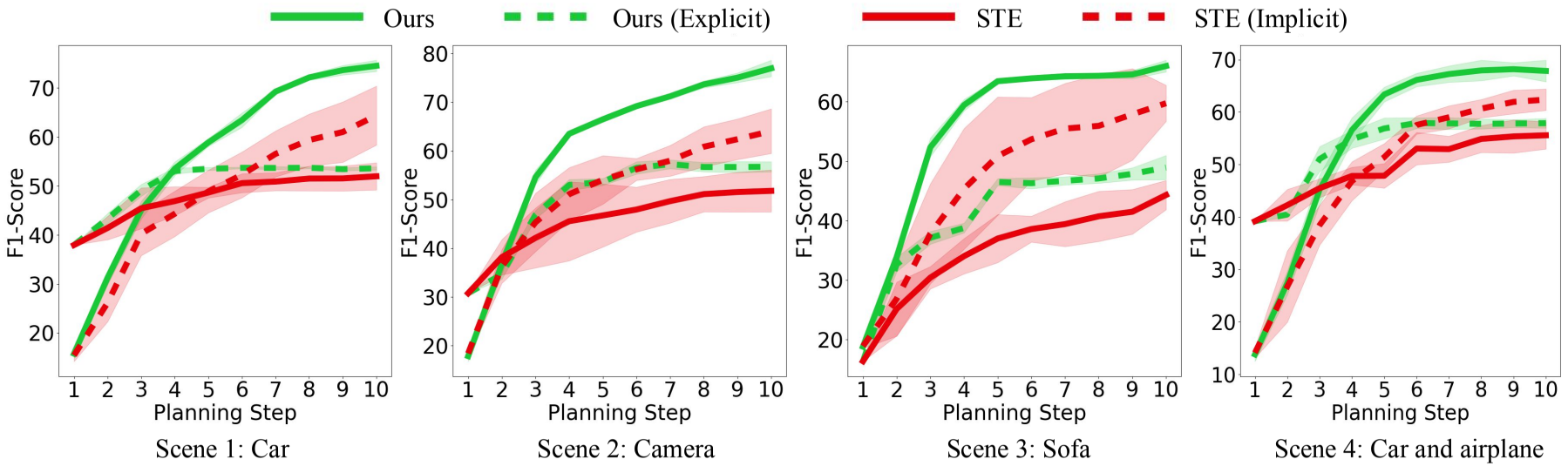


Figure 5.6: Comparison of our approach against the semantic-targeted active explicit reconstruction system *STE* [208]. Dashed lines denote variants cross-validating the measurements collected by one active reconstruction system with the mapping method of the other. The same color indicates mapping using the same measurements. The results confirm that our STAIR pipeline achieves superior performance compared to the explicit baseline. The performance gain originates from the implicit neural representation used in our approach and our utility function for finding more informative measurements.

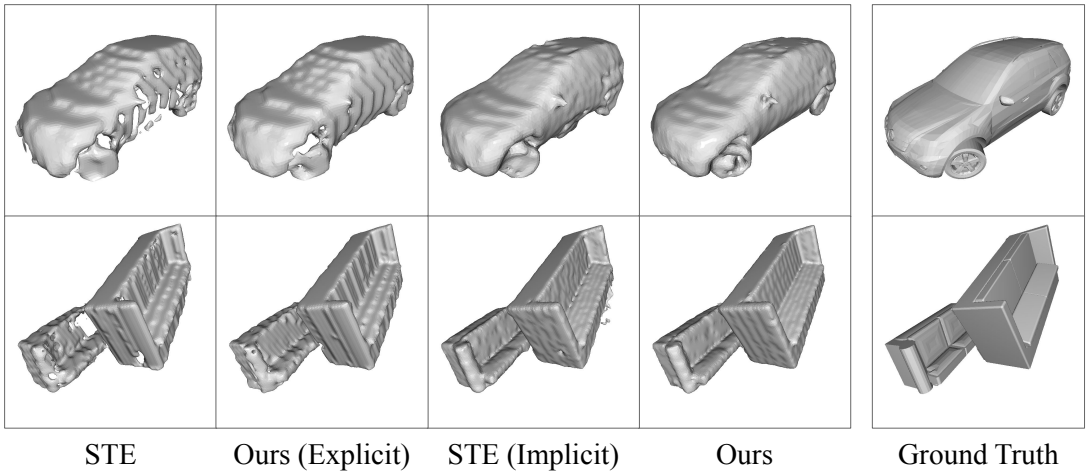


Figure 5.7: Comparison of final mesh reconstructions. The meshes extracted from explicit map representations are limited by the discrete representation, containing holes and non-smooth surfaces. The implicit neural representation used in our approach results in better mesh quality, due to its continuous representation capabilities.

quality. This indicates that our semantic-targeted view planning based on dense semantic and uncertainty rendering enables finding more informative viewpoints to reconstruct objects of interest. Figure 5.7 visualizes the final extracted meshes using the four methods. Meshes extracted from our implicit neural representation show complete surfaces with more details compared to those from explicit maps.

5.2.4 Ablation Study

The final experiment justifies our design choice for the utility function introduced in Section 5.1.3. We show that an exploration term is necessary for semantic-targeted view planning in an unknown environment. For this purpose, we design a challenging scene, as shown in Figure 5.8, where two objects of interest (chairs) are separated by other objects. We start from the top viewpoint, from which only one chair is seen and the other one is occluded. We compare the planning approach using the exploitation-only score in Equation (5.13), i.e., $\varepsilon = 0.0$, and our proposed utility function in Equation (5.15) with ε values of 0.2, 0.5, and 0.8 to investigate the influence of varying the exploration term proportion.

Figure 5.8 compares the reconstruction performance of both rendering and mesh quality. Semantic-targeted view planning without exploration focuses only on already detected objects of interest. As a result, this planning strategy does not explore the unknown environment to find other potential objects of interest in the scene, leading to inferior overall reconstruction performance. In contrast, our approach trades off between exploring the unknown environment and exploiting information about objects of interest as they are discovered. The results indicate

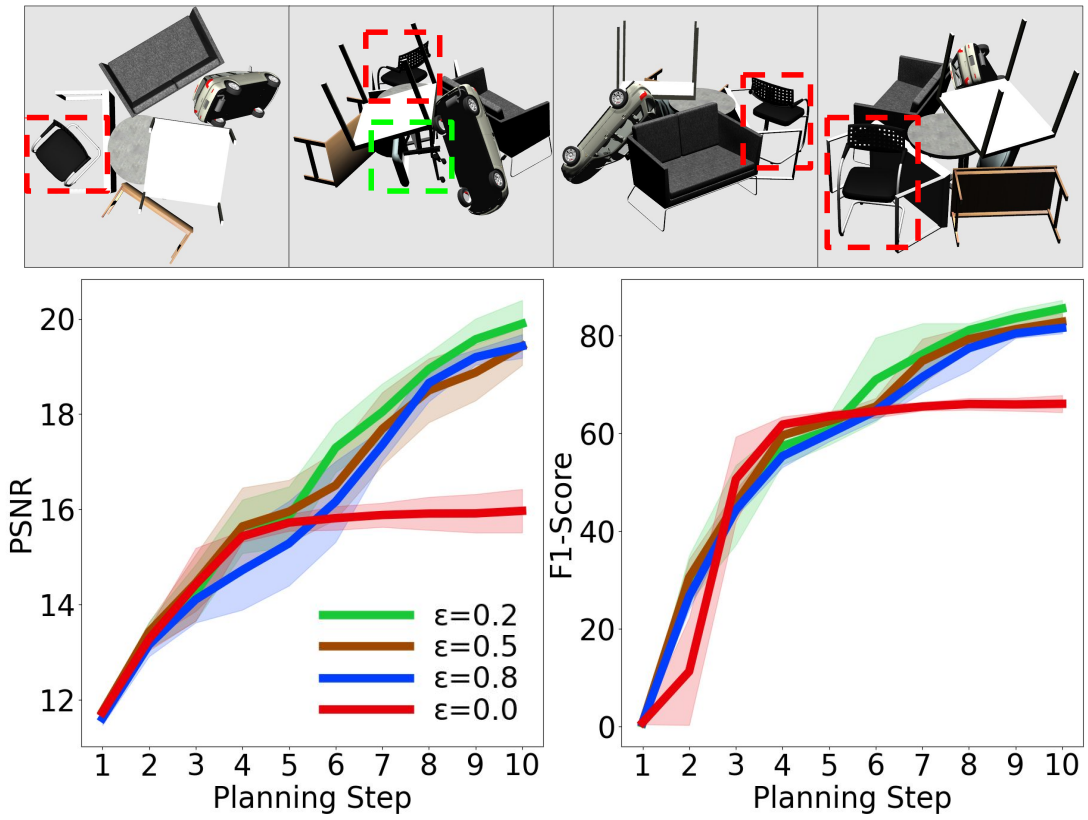


Figure 5.8: Top row: Test scene seen from different perspectives. One object of interest (red bounding box) can be easily detected; however, the second object of interest (green bounding box) is severely occluded by other objects and can only be observed from particular viewpoints. Bottom row: Semantic-targeted view planning using an exploitation term alone ($\varepsilon = 0.0$) cannot explore to find both objects of interest. In contrast, our utility function balances between exploitation and exploration, leading to better active reconstruction performance in this challenging situation.

that a small exploration term is sufficient to achieve such behavior, while up-weighting exploration deteriorates semantic-targeted view planning performance.

5.3 Related Work

Our approach lies at the intersection of active reconstruction using semantics and implicit neural representations. By integrating task-driven view planning with learned implicit models that encode both geometry and semantics, we bridge these two research areas. In this section, we provide an overview of related work in both fields.

5.3.1 Semantic-Targeted Active Explicit Reconstruction

Semantic understanding is crucial for many autonomous robotic tasks in unknown environments. Recent advancements in deep learning-based semantic segmentation facilitate the seamless integration of semantic understanding onboard robotic systems [62]. In the context of active reconstruction, several works propose integrating semantics into explicit maps to enable semantic-targeted view planning.

Papatheodorou *et al.* [127] employ a coarse occupancy voxel map to model the background for exploring unknown environments. Upon detecting objects belonging to predefined semantic classes of interest, they reconstruct these objects in detail using an adaptive-resolution octree-based signed distance function. Burusa *et al.* [17] estimate the expected information gain based on the confidence score of a voxel belonging to interesting semantic classes and use it for view planning. To address occlusion challenges in active reconstruction, Lehnert *et al.* [90] design a 3D camera array to obtain multiple measurements from different perspectives. The objects of interest detected in each measurement are used to calculate the gradient, indicating the most likely direction of movement to better observe them. Similar to our problem setup, Zaenker *et al.* [208] propose a semantic-targeted active explicit reconstruction system based on occupancy voxel maps and apply it to reconstruct fruits in agricultural robotics applications. To guide targeted NBV planning, they assign higher utility for candidate viewpoints that observe more unknown voxels close to already detected objects of interest.

Our approach shares the same idea of using semantic information to conduct view planning toward objects of interest. However, different from previous works that rely on discrete explicit maps, we exploit recent advances in implicit neural representations to improve the reconstruction quality.

5.3.2 Active Implicit Reconstruction

Implicit neural representations are a powerful tool for 3D reconstruction due to their continuous representation capabilities. As discussed in Chapter 4.3.3, recent work has explored these benefits in active reconstruction settings.

Pan *et al.* [124] model the radiance field as a Gaussian distribution and actively collect images by evaluating the reduction of uncertainty assuming new inputs at candidate viewpoints. Exploiting fast rendering of Instant-NGP [110], Sünderhauf *et al.* [177] train an ensemble of NeRF models for a single scene and measure uncertainty as the variance of the ensemble's prediction, which is used to conduct NBV planning. Similar to our approach, Lee *et al.* [89] use entropy of density distribution along rendering rays as the uncertainty measure to identify areas with low reconstruction quality. Leveraging the differentiability of the implicit neural representations, Yan *et al.* [203] optimize viewpoint generation

toward areas of high uncertainty. Following a different paradigm, Pan *et al.* [123] utilize a prediction network to predict the number of viewpoints required to reconstruct a specific unknown object using NeRF, allowing for one-shot viewpoint sequence generation without online replanning.

Our work follows these lines by using implicit neural representations for active reconstruction. Unlike previous methods that uniformly reconstruct a scene or an object, our approach integrates semantic understanding into an implicit neural representation to achieve semantic-targeted active implicit reconstruction.

5.3.3 Semantics in Implicit Neural Representations

Recent works propose lifting 2D semantic information into 3D to generate a consistent semantic field. These methods exploit the multi-view consistency during implicit neural representations training. Zhi *et al.* [217] extend vanilla NeRF to jointly predict the semantics of each sampling point along with color and density values. Their results show multi-view consistent and smooth semantic rendering at novel viewpoints, even given sparse or noisy 2D semantic labels as supervision signals. Siddiqui *et al.* [156] and Bhalgat *et al.* [8] further incorporate instance segmentation into implicit neural representations. Different from lifting 2D semantics into 3D, Vora *et al.* [189] directly train a 3D network to convert a learned density field into a semantic field, which generalizes across scenes.

In contrast to previous approaches for generating semantic implicit neural representations, Kelly *et al.* [76] use semantic information to train NeRFs in a targeted way. To reconstruct objects of interest in the scene at a higher quality, they propose a denser sampling of training examples around these objects based on semantic segmentation. DietNeRF [67] proposes a semantic consistency loss to regularize rendering from arbitrary viewpoints, encouraging consistent high-level semantics. This additional loss alleviates the degenerate performance commonly observed in NeRF training with sparse measurements.

While semantics offer rich scene understanding capabilities in implicit neural representations, they have not yet been applied to active implicit reconstruction problems. We bridge this gap by introducing an approach for semantic-targeted active reconstruction based on implicit neural representations. Our approach is applicable to similar problems tackled by current methods using active explicit reconstruction to target objects of interest in unknown environments [17,127,208]. However, we exploit the advantages of underlying implicit neural representations to further improve the reconstruction quality.

5.4 Conclusion

This chapter presents STAIR, a novel approach for semantic-targeted active implicit reconstruction in unknown environments. We integrate semantic understanding into implicit neural representations to guide view planning toward objects of interest. By jointly modeling semantic rendering and uncertainty estimation, STAIR actively selects informative viewpoints that prioritize regions containing objects of interest, rather than treating all areas equally.

Our active planning experiments show that STAIR outperforms both standard implicit reconstruction baselines that ignore semantics and a semantic-targeted approach based on explicit map representations. These results support our motivation to combine implicit neural representations with semantic-targeted view planning to enhance the reconstruction quality of target objects. Our ablation study also highlights the importance of exploration behaviors in planning, particularly in complex and cluttered scenes, where occluded or partially visible target objects may otherwise be overlooked.

Despite its effectiveness for semantic-targeted reconstruction, similar to Chapter 4, STAIR still requires computationally intensive dense sampling for volume rendering. In the next chapter, we propose an active perception approach for robot mapping to address the challenge of photorealistic reconstruction of unknown environments using a more efficient learning-based map representation.

Chapter 6

Active Scene Reconstruction Using Gaussian Splatting

In this chapter, we focus on online active reconstruction of unknown scenes under a limited budget, e.g., mission time, where the goal is to efficiently obtain a photorealistic 3D representation by actively positioning the robot with its camera online during a mission. Traditional active scene reconstruction approaches mainly rely on conventional map representations such as voxel grids, meshes, or point clouds [11, 80, 152, 160, 161, 210, 219]. However, these methods often do not deliver high-fidelity reconstruction results due to their sparse representations. Recent advances in implicit neural representations, e.g., NeRFs [107], have attracted significant research interest for their accurate dense scene reconstruction capabilities and low memory footprints. In the context of active reconstruction, several emerging works [38,124,139,203] incorporate uncertainty estimation in NeRFs and exploit it to guide view planning. While these approaches demonstrate promising results, the rather costly dense sampling for volume rendering procedure during online incremental mapping poses limitations for NeRF-based active scene reconstruction, as we have discussed in our approaches previously introduced in Chapter 4 and Chapter 5.

Dense reconstruction using Gaussian splatting (GS) [78] offers a promising alternative to NeRF-based approaches, addressing rendering inefficiencies while preserving representation capabilities. GS explicitly models scene properties through Gaussian primitives and utilizes efficient differentiable rasterization to achieve novel view synthesis. Its fast map updates and explicit structure make it well-suited for online incremental mapping. Building on these strengths, we adopt GS for active scene reconstruction in this chapter.

While showing promising online incremental mapping results [75, 104], incorporating GS into an active scene reconstruction pipeline presents significant challenges. First, active reconstruction often requires evaluating the reconstruction

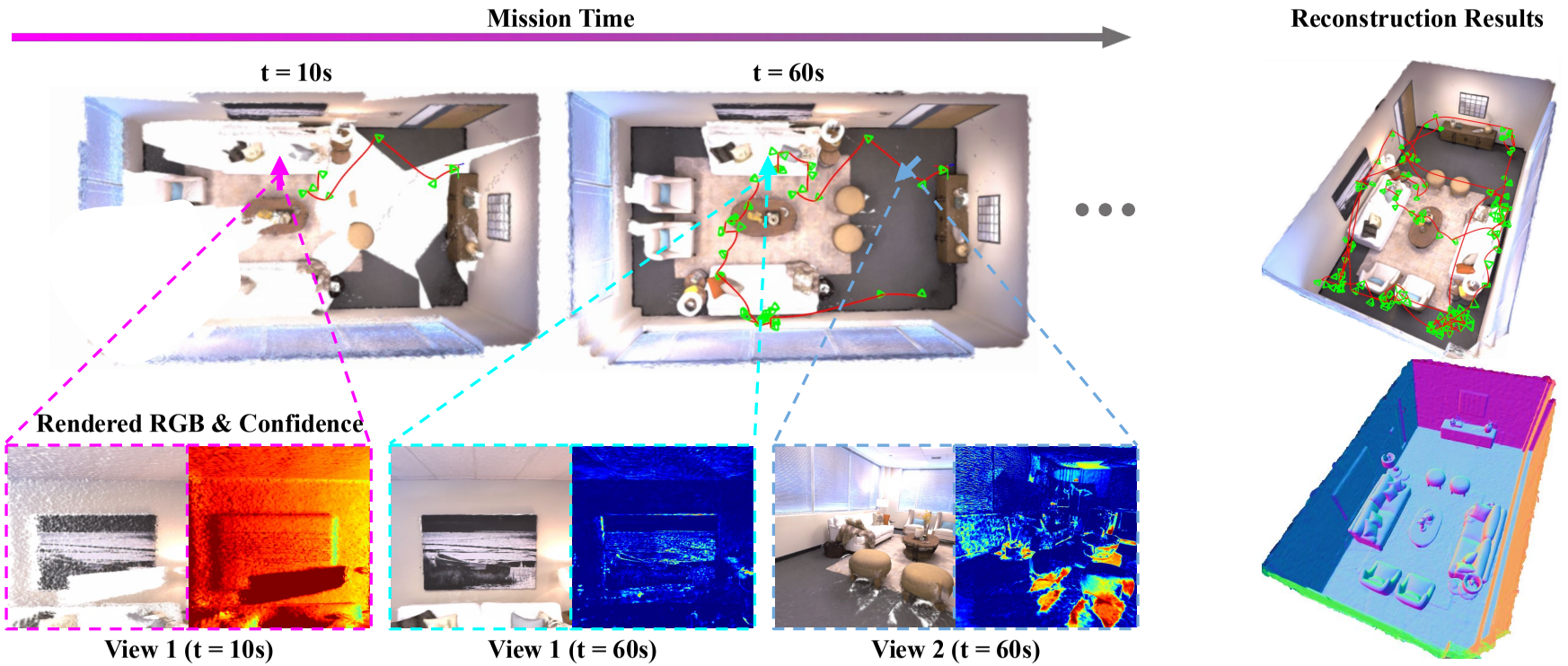


Figure 6.1: Our approach actively reconstructs an unknown scene. We illustrate the reconstruction progress over mission time, displaying planned camera viewpoints (green pyramids) and paths (red lines). We present examples of RGB and confidence images (redder color indicates lower confidence) rendered at the same viewpoint at different mission times (magenta and cyan arrows) and at two distinct viewpoints at the same mission time (cyan and blue arrows). By integrating confidence modeling into the Gaussian splatting pipeline, our approach enables targeted view planning to build a high-fidelity GS map. The complete camera path and final reconstruction results, including RGB rendering and surface mesh, are visualized on the right.

quality to guide view planning. However, this is difficult without ground-truth information at novel viewpoints. Second, the Gaussian primitives represent only occupied space, making it hard to distinguish between unknown and free space, which are important for exploration and path planning.

The key contribution of this chapter is addressing these challenges with our GS-based active scene reconstruction approach, ActiveGS, as illustrated in Figure 6.1. To tackle the first challenge, we propose a simple yet effective confidence modeling technique for Gaussian primitives based on viewpoint distribution, enabling view planning for inspecting under-reconstructed surfaces. For the second challenge, we combine the GS map with a conventional coarse voxel map, exploiting the strong representation capabilities of GS for scene reconstruction with the spatial modeling strengths of voxel maps for exploration and path planning.

We make the following three claims:

1. Our ActiveGS approach achieves superior reconstruction performance compared to state-of-the-art NeRF-based approach and GS-based baselines.
2. Our confidence modeling for Gaussian primitives enables informative viewpoint evaluation and targeted inspection around under-reconstructed surfaces, further improving mission efficiency and reconstruction quality.
3. We validate our approach in different synthetic indoor scenes and in a real-world scenario using a UAV.

6.1 Our Approach to Active Reconstruction Using Gaussian Splatting

We introduce ActiveGS, a novel approach for active scene reconstruction using GS. An overview of our approach is shown in Figure 6.2. Our goal is to reconstruct an unknown scene using a mobile robot, e.g., a UAV, equipped with an onboard RGB-D camera. Given posed RGB-D measurements as input, we maintain a coarse voxel map to model the spatial occupancy and incrementally train a GS map for high-fidelity scene reconstruction. To actively guide view planning in a targeted manner, we propose using our confidence modeling technique in the GS map together with information about unexplored regions in the voxel map as the basis for planning. Our approach alternates between mapping and planning steps until a predefined mission time is reached.

6.1.1 Hybrid Map Representation

Assuming a bounding box of the scene to be reconstructed is given, we uniformly divide the enclosed space into voxels, forming our voxel map $\mathcal{V}$, where each voxel

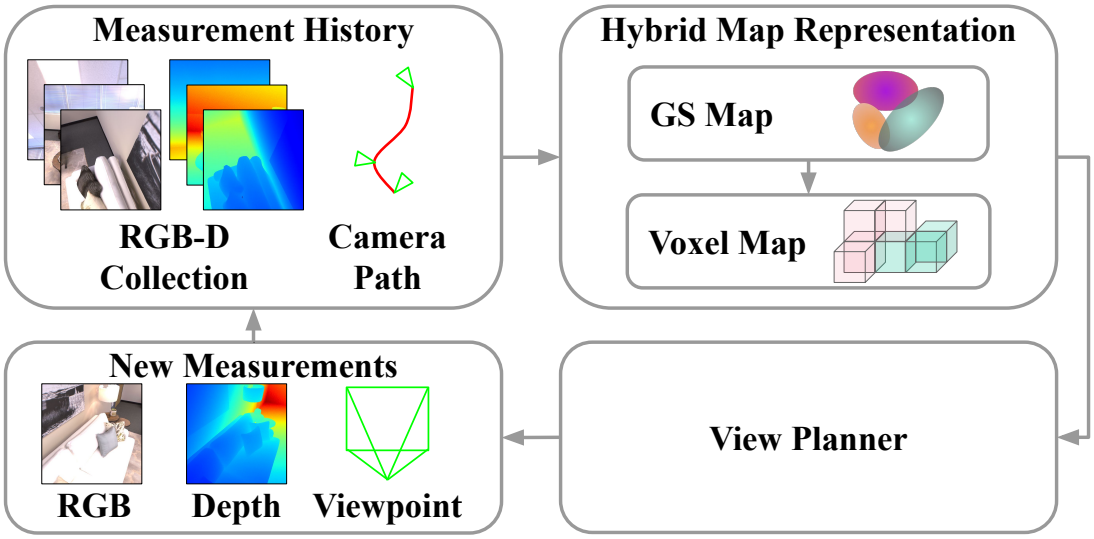


Figure 6.2: An overview of the proposed ActiveGS approach. Our hybrid map representation consists of a GS map for high-fidelity scene reconstruction with a coarse voxel map for exploration and path planning. Our view planner leverages unexplored regions in the voxel map for exploration and low-confidence Gaussian primitives for targeted inspection, collecting informative measurements at planned viewpoints for map updates. We iterate between the map update and view planning steps until the preallocated mission time is reached.

$v_i \in \mathcal{V}$ represents the volume occupancy probability in the range of $[0\,,1]$.

Our GS map is based on Gaussian surfel [29], a state-of-the-art GS representation that leverages 2D Gaussian primitives. The GS map $\mathcal{G}$ comprises a collection of Gaussian primitives. Each primitive $\mathbf{g}_i \in \mathcal{G}$ is defined by its parameters $\mathbf{g}_i = (\mathbf{x}_i\,, \mathbf{q}_i\,, \mathbf{s}_i,\ \mathbf{c}_i\,, o_i\,, k_i)$, where $\mathbf{x}_i \in \mathbb{R}^3$ denotes the position of the primitive center; $\mathbf{q}_i \in \mathbb{R}^4$ is its rotation in the form of a quaternion; $\mathbf{s}_i = [s_i^x\,, s_i^y] \in \mathbb{R}^2_+$ are the scaling factors along the two axes of the primitive; $\mathbf{c}_i \in [0\,,1]^3$ represents the RGB color; $o_i \in [0\,,1]$ is the opacity; and $k_i \in \mathbb{R}_+$ is the confidence score introduced later in Section 6.1.3. The distribution of the Gaussian primitive $\mathbf{g}_i$ in world coordinate is defined as:

$$\mathcal{N}(\mathbf{x}; \mathbf{x}_i, \mathbf{\Sigma}_i) = \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{x}_i)^\top \mathbf{\Sigma}_i^{-1}(\mathbf{x}-\mathbf{x}_i)\right), \tag{6.1}$$

where $\mathbf{\Sigma}_i = \mathbf{R}(\mathbf{q}_i)\,\mathrm{diag}\left((s_i^x)^2, (s_i^y)^2, 0\right)\,\mathbf{R}(\mathbf{q}_i)^\top$ is the covariance matrix, with the rotations matrix $\mathbf{R}(\mathbf{q}_i) \in SO(3)$ derived from the corresponding quaternion $\mathbf{q}_i$, and $\mathrm{diag}(\,\cdot\,)$ indicating a diagonal matrix with the specified diagonal elements. The normal of the Gaussian primitive can be directly obtained from the last column of the rotation matrix as $\mathbf{n}_i = \mathbf{R}(\mathbf{q}_i)_{:,3}$.

Given the GS map, we can render the color image $\mathbf{C} \in \mathbb{R}^{H\times W\times 3}$, depth image $\mathbf{D} \in \mathbb{R}^{H\times W}$, normal image $\mathbf{N} \in \mathbb{R}^{H\times W\times 3}$, opacity image $\mathbf{O} \in \mathbb{R}^{H\times W}$, and confidence image $\mathbf{K} \in \mathbb{R}^{H\times W}$ at a viewpoint using the differentiable rasterization

pipeline [29], where H and W denote the image height and width. With a slight abuse of notation, we denote the corresponding rendering function for a pixel $\mathbf{u}$ as $\mathbf{C}(\mathbf{u})$, $D(\mathbf{u})$, $\mathbf{N}(\mathbf{u})$, $O(\mathbf{u})$, and $K(\mathbf{u})$, respectively. Without loss of generality, the rendering process is formulated as:

$$O(\mathbf{u}) = \sum_{i=1}^{n} w_i \,, M(\mathbf{u}) = \sum_{i=1}^{n} w_i m_i \,, \tag{6.2}$$

$$w_i = T_i \alpha_i \,, T_i = \prod_{j<i} (1 - \alpha_j) \,, \alpha_i = \mathcal{N}(\mathbf{u}; \mathbf{u}_i, \mathbf{\Sigma}'_i)\, o_i \,, \tag{6.3}$$

where $M \in \{\mathbf{C}, D, \mathbf{N}, K\}$, and $m_i \in \{\mathbf{c}_i, d_i, \mathbf{n}_i, k_i\}$ is the corresponding modality feature, with d_i being the distance from the viewpoint center to the intersection point of the camera ray and the Gaussian primitive $\mathbf{g}_i$; w_i indicates the rendering contribution of $\mathbf{g}_i$ to pixel $\mathbf{u}$; $\mathbf{\Sigma}'_i$ and $\mathbf{u}_i$ are the primitive's covariance matrix and center projected onto the image space [223]. For more technical details about the rendering process, please refer to Gaussian surfel [29].

6.1.2 Incremental Mapping

We collect measurements captured at planned viewpoints and incrementally update our map representation. Given the current RGB image $\mathbf{C}^\star$ and depth image $\mathbf{D}^\star$ measurement, we generate a per-pixel point cloud using known camera parameters. We then update our voxel map $\mathcal{V}$ probabilistically based on the new point cloud, following OctoMap [59].

For the GS map update, we first add Gaussian primitives to $\mathcal{G}$ where needed. To this end, we render the color image $\mathbf{C}$, depth image $\mathbf{D}$, and opacity image $\mathbf{O}$ at the current camera viewpoint. We calculate a binary mask to identify the pixels in the new measurement that should be considered for densifying the GS map:

$$\mathbf{B} = (\mathbf{O} < 0.5) \vee (\text{avg}(|\mathbf{C} - \mathbf{C}^\star|) > 0.5) \vee ((\mathbf{D} - \mathbf{D}^\star) > \lambda \mathbf{D}^\star) \,, \tag{6.4}$$

where $\text{avg}(\,\cdot\,)$ is the channel-wise averaging operation to calculate per-pixel color error image, and λ is a constant accounting for depth sensing noise, set to 0.05 in our pipeline. This mask indicates areas where opacity is low, color rendering is inaccurate, or new geometry appears in front of the current depth estimate, signalling the need for new Gaussian primitives. We spawn new Gaussian primitives by unprojecting pixels on these areas into 3D space, with initial parameters defined by the corresponding point cloud position, pixel color, and normal estimated by applying central differencing on the bilateral-filtered depth image [114], which helps mitigate noise contained in the depth sensing. We also set scale values to 1 cm, opacity value to 0.5, and confidence value to 0.

At each mapping step, we train our GS map $\mathcal{G}$ using all collected RGB-D measurements for 10 iterations. Specifically, for each iteration, we select the 3

most recent frames and 5 random frames from the measurement history. The loss for a frame $\{\hat{\mathbf{C}}, \hat{\mathbf{D}}\}$ in the training batch is formulated as the weighted sum of individual loss terms:

$$\mathcal{L} = \lambda_c \mathcal{L}_c + \lambda_d \mathcal{L}_d + \lambda_n \mathcal{L}_n \,, \tag{6.5}$$

where the photometric loss $\mathcal{L}_c = \mathrm{L}_1(\mathbf{C}, \hat{\mathbf{C}})$ and the depth loss $\mathcal{L}_d = \mathrm{L}_1(\mathbf{D}, \hat{\mathbf{D}})$ are both calculated using the L_1 distance. We formulate the loss related to normal estimate as $\mathcal{L}_n = \mathrm{D}_{\cos}(\mathbf{N}, \widetilde{\mathbf{N}}) + \mathrm{TV}(\mathbf{N})$, which consists of the cosine distance $\mathrm{D}_{\cos}$ between the rendered normal image and the normal image $\widetilde{\mathbf{N}}$ derived from the rendered depth image [29], along with the total variation TV loss [147] to enforce smooth normal rendering between neighboring pixels. λ_c, λ_d, and λ_n are the weights for the corresponding loss terms. Note that the training process involves only a subset of the Gaussian primitive parameters $(\mathbf{x}_i, \mathbf{q}_i, \mathbf{s}_i, \mathbf{c}_i, o_i)$, while the modeling of non-trainable k_i is introduced in Section 6.1.3.

After every 5 mapping steps, we perform a visibility check on all Gaussian primitives and delete those invisible from all history viewpoints to compact the GS map. We consider a Gaussian primitive visible from a viewpoint if at least one pixel rendered in that view receives its rendering contribution greater than a threshold, as defined in Equation (6.3). Unlike previous works utilizing density control during offline training [29,61,78], our approach adds necessary primitives and removes invisible ones during online missions, achieving computationally efficient scene reconstruction.

6.1.3 Confidence Modeling for Gaussian Primitives

A Gaussian primitive can be effectively optimized if observed from different viewpoints. Based on this insight, we derive the confidence of a Gaussian primitive from the spatial distribution of its visible viewpoints in the measurement history. Specifically, we connect the Gaussian center $\mathbf{x}_i$ to the viewpoint center $\mathbf{x}_{\mathbf{p}_j}$, denoted as $\mathbf{d}_{ij} = \mathbf{x}_{\mathbf{p}_j} - \mathbf{x}_i = d_{ij}\mathbf{v}_{ij}$, where d_{ij} is the distance and $\mathbf{v}_{ij}$ is the normalized view direction, with $j \in \mathcal{S}(\mathbf{g}_i)$ and $\mathcal{S}$ being the index set of viewpoints from which the Gaussian primitive $\mathbf{g}_i$ is observed. We formulate the confidence k_i as:

$$k_i = \gamma_i \exp(\beta_i) \,, \tag{6.6}$$

$$\gamma_i = \sum_{j \in \mathcal{S}(\mathbf{g}_i)} \left(1 - \frac{d_{ij}}{d_{\text{far}}}\right) \mathbf{n}_i \cdot \mathbf{v}_{ij} \,, \tag{6.7}$$

$$\beta_i = 1 - \|\boldsymbol{\mu}_i\|_2 \tag{6.8}$$

$$\boldsymbol{\mu}_i = \frac{1}{|\mathcal{S}(\mathbf{g}_i)|} \sum_{j \in \mathcal{S}(\mathbf{g}_i)} \mathbf{v}_{ij} \,, \tag{6.9}$$

where γ_i accounts for distance-weighted cosine similarity between the Gaussian primitive's normal $\mathbf{n}_i$ and view direction $\mathbf{v}_{ij}$, with d_{far} as the maximum depth

sensing range. Note that we increase the impact of viewpoints that are closer to the Gaussian primitive's center or provide view directions similar to its normal. β_i measures the dispersion of directions from which $\mathbf{g}_i$ is observed, with β_i closer to 0 indicating similar view directions. Our confidence formulation assigns higher confidence to Gaussian primitives densely observed from viewpoints with varying angles, whereas lower confidence to those with sparse and similar measurements.

6.1.4 Viewpoint Utility Formulation

Active scene reconstruction requires both exploration, to cover unknown areas, and exploitation, to closely inspect under-reconstructed surfaces. In our approach, we combine utility derived from the voxel map for exploration and the GS map for exploitation, enabling these behaviors effectively.

A candidate viewpoint v_i^c is defined by its 3D position together with yaw and pitch angles in our approach. To simplify path planning, we constrain the positions to a discrete lattice placed at the centers of all free voxels. We follow existing active scene exploration methods [11, 64, 122, 152, 219] and define the exploration utility based on the number of unexplored voxels visible from a candidate viewpoint. Without relying on computationally expensive ray-casting operations, we leverage the efficient rendering capabilities of the GS map to determine voxel visibility. We achieve this by checking whether the projected depth of the in-view voxel centers in the camera coordinate is smaller than the corresponding depth value in the rendered depth from the GS map.

Combining unexplored region information in the voxel map and confidence rendering from the GS map, we define the utility of a candidate viewpoint v_i^c as:

$$\psi_{\text{view}}(v_i^c) = \phi\, \psi_{\mathcal{V}}(v_i^c) + \psi_{\mathcal{G}}(v_i^c)\,, \tag{6.10}$$

where $\psi_{\mathcal{V}}(v_i^c) = \frac{N_u(v_i^c)}{|\mathcal{V}|}$ is the exploration utility, defined by the ratio of the number of visible unexplored voxels $N_u(v_i^c)$ to the total number of voxels in the voxel map; $\psi_{\mathcal{G}}(v_i^c) = -\text{mean}(\mathbf{K}_i)$ is the exploitation utility, calculated as the negative mean of the confidence image $\mathbf{K}_i$ rendered at v_i^c following Equation (6.2); and ϕ is the exploration weight.

6.1.5 Viewpoint Sampling and Evaluation

Our viewpoint sampling strategy involves two types of candidate viewpoints. First, we randomly sample N_{random} candidate viewpoints within a specified range around the current viewpoint. However, relying solely on random local sampling can lead to local minima. To address this, we introduce additional candidate viewpoints based on regions of interest defined in the voxel map. We begin by identifying frontier voxels [202] and add them to our regions of interest set $\mathcal{R}$.

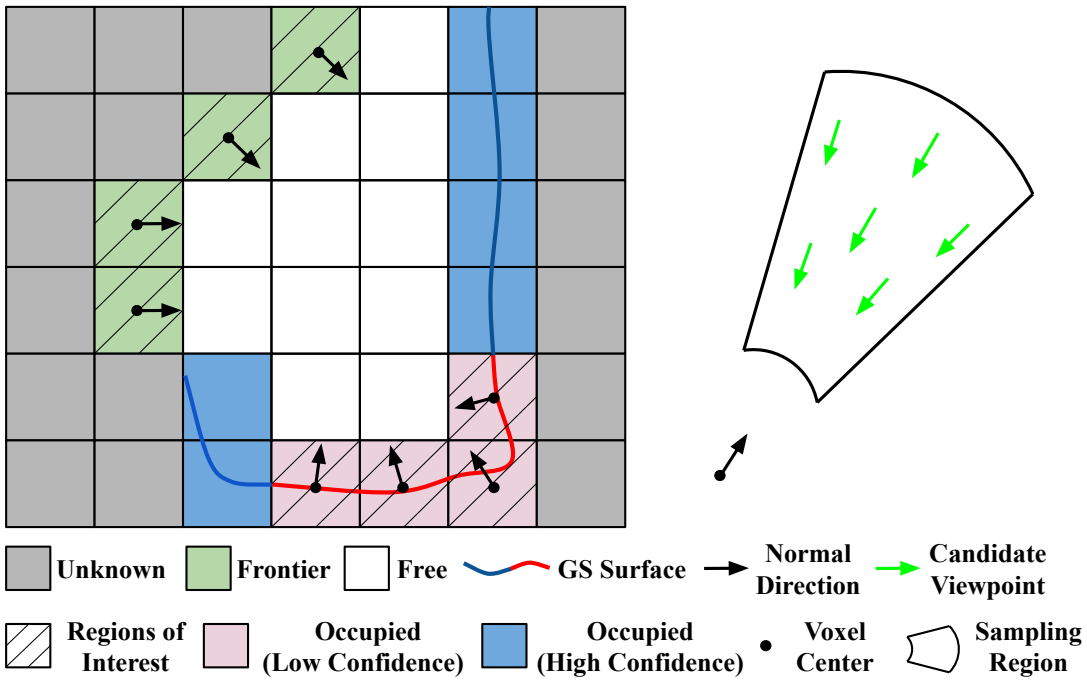


Figure 6.3: We show a 2D case of our regions-of-interest-based candidate viewpoint generation. We define regions of interest as voxels containing low-confidence Gaussian primitives and frontier voxels. Normals for low-confidence voxels are generated by averaging the normals of low-confidence primitives, while frontier voxel normals are calculated using average vectors to neighboring free voxels. Given the voxel centers and directional normals, we generate candidate viewpoints within the sampling region, as illustrated on the right.

By explicitly modeling the confidence of each Gaussian primitive, we can identify and also include voxels containing low-confidence Gaussian primitives in $\mathcal{R}$. Inspired by previous work [80], we define normals for each voxel in $\mathcal{R}$ to indicate the most informative viewing direction. For voxels with low-confidence Gaussian primitives, this is simply the average normal of these Gaussian primitives. The normal of frontier voxels is determined by finding their neighboring free voxels and calculating the average directional vector from the frontier voxel to these neighbors. To generate candidate viewpoints based on regions of interest, we create a fixed number of candidate viewpoints within a cone defined by the minimum and maximum sampling distances from the center of each voxel in $\mathcal{R}$ and the maximum angular difference relative to its normal. Starting from the closest voxel, we continue outward until we have collected up to N_{ROI} viewpoints in free space. We illustrate the sampling process in Figure 6.3.

We evaluate the utility of all candidate viewpoints following Equation (6.10). We use the A^* algorithm [52] to find the shortest traversable path from the current viewpoint position to all candidate viewpoint positions. Taking travel distance

into account, we select the next best viewpoint v^* by:

$$v^* = \arg\max_{v_i^c} \left(\frac{\psi_{\text{view}}(v_i^c)}{\sum_{i=1}^{N_{\text{total}}} \psi_{\text{view}}(v_i^c)} - \delta \frac{U_{\text{path}}(v_i^c, v_{\text{current}})}{\sum_{i=1}^{N_{\text{total}}} U_{\text{path}}(v_i^c, v_{\text{current}})} \right), \tag{6.11}$$

where $N_{\text{total}} = N_{\text{random}} + N_{\text{ROI}}$; U_{path} is the travel distance from the current viewpoint v_{current} to a candidate viewpoint positions; and δ is a weighting factor for the travel cost.

6.2 Experimental Evaluation

Our experimental results support our three claims: (i) we show that our ActiveGS pipeline outperforms state-of-the-art NeRF-based and GS-based active scene reconstruction methods; (ii) we show that our confidence modeling of Gaussian primitives enables informative viewpoint evaluation and targeted candidate viewpoint generation, improving reconstruction performance; and (iii) we validate our approach across different scenes in simulation as well as in a real-world scenario to demonstrate its practical applicability.

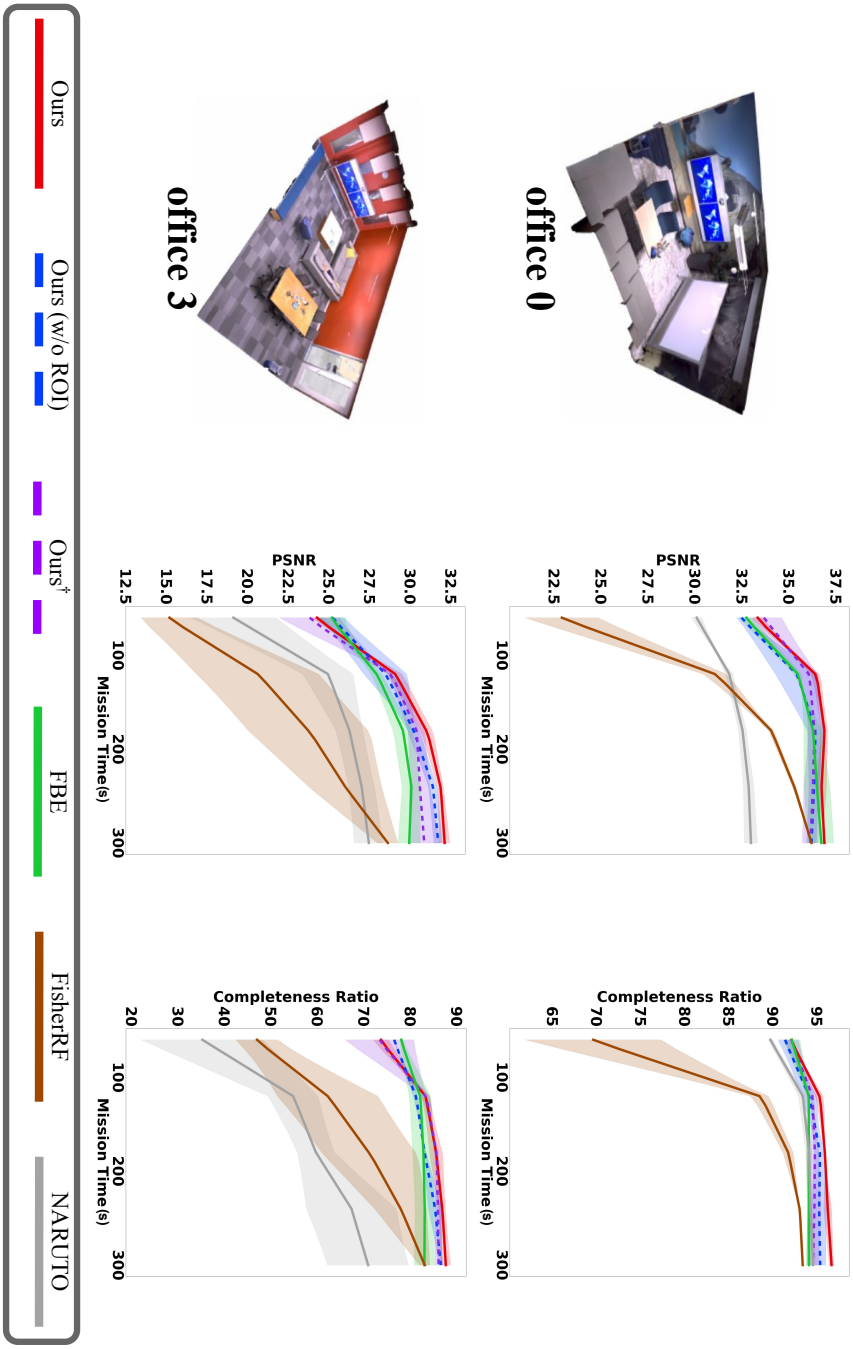
6.2.1 Implementation Details

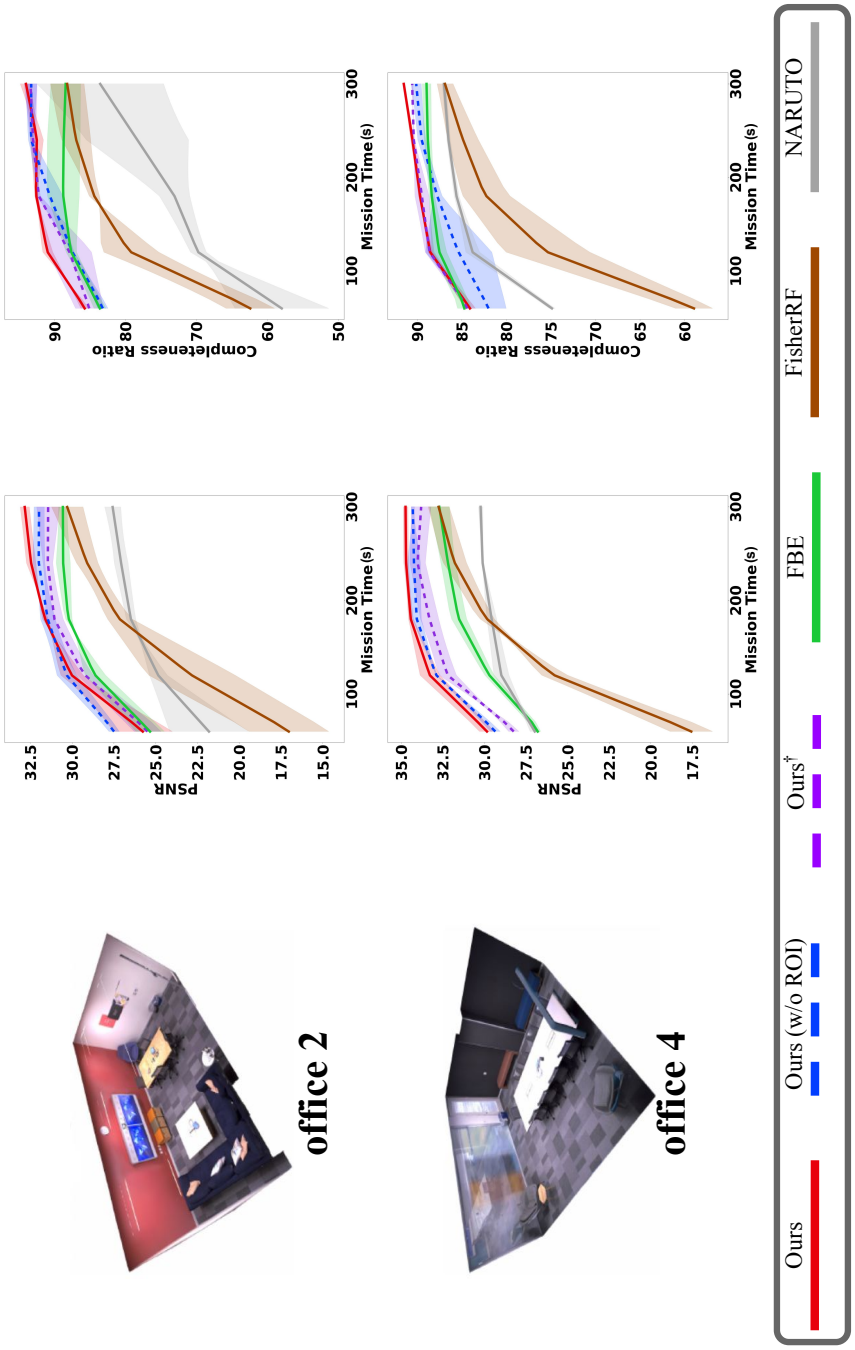
We use a voxel size of 20 cm × 20 cm × 20 cm for the voxel map and set the loss weights in Equation (6.5) as: $\lambda_c = 1.0$, $\lambda_d = 0.8$, and $\lambda_n = 0.1$. For visibility checks, a minimum rendering contribution threshold of 0.3 is applied. We set the exploration weight $\phi = 1000$ in Equation (6.10) to encourage exploratory behavior during the initial phase of an online mission, and travel costs in Equation (6.11) are weighted by $\delta = 0.5$. We consider $N_{\text{total}} = 100$ candidate viewpoints, including up to $N_{\text{ROI}} = 30$ samples around regions of interest and $N_{\text{random}} = N_{\text{total}} - N_{\text{ROI}}$ random samples generated within 0.5 m distance to the current viewpoint.

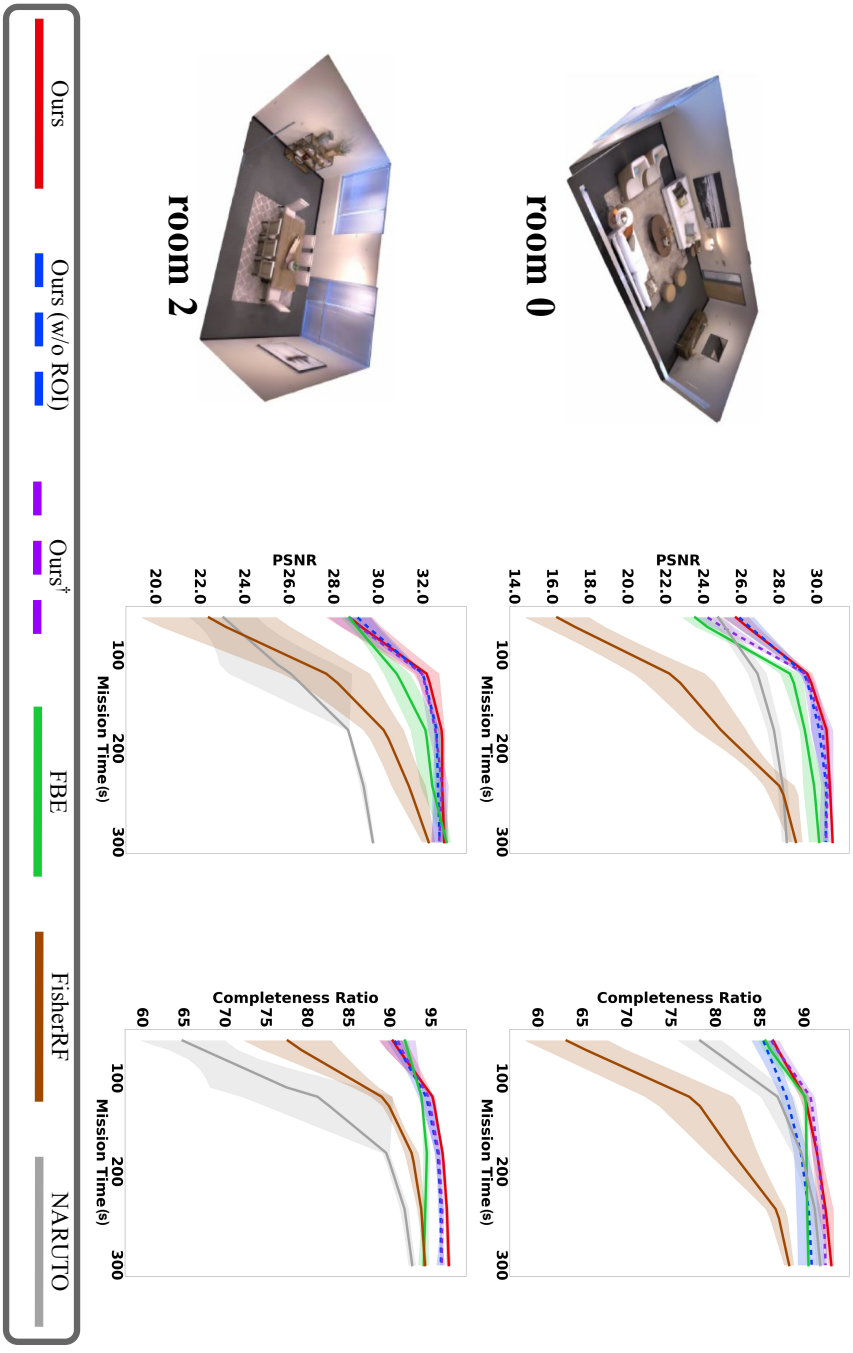
We test our implementation on a desktop PC with an Intel Core i9-10940X CPU and an NVIDIA RTX A5000 GPU. In this setup, one mapping and planning steps take approximately 1 s and 0.5 s, respectively. The whole pipeline consumes $4 - 5$ GB GPU RAM during an online mission, with approximately 10% allocated to the voxel map update.

6.2.2 Simulation Experiments

We conduct our simulation experiments using the Habitat simulator [149] and the Replica dataset [169]. The experiments utilize an RGB-D camera with a field of view of $[60°, 60°]$ and a resolution of 512 × 512 pixels. The depth sensing ranges from $[0.1, 5.0]$ m and is subject to Gaussian noise with a standard deviation that increases linearly with depth, $\sigma = 0.01d$, where d is the depth value in meters.


office 0
office 3
PSNR
Completeness Ratio
Mission Time(s)
Ours
Ours (w/o ROI)
Ours†
FBE
FisherRF
NARUTO


Completeness Ratio
Mission Time(s)
PSNR
office 2
office 4
Ours
Ours (w/o ROI)
Ours†
FBE
FisherRF
NARUTO


room 0
room 2
PSNR
Completeness Ratio
Mission Time(s)
Ours
Ours (w/o ROI)
Ours†
FBE
FisherRF
NARUTO

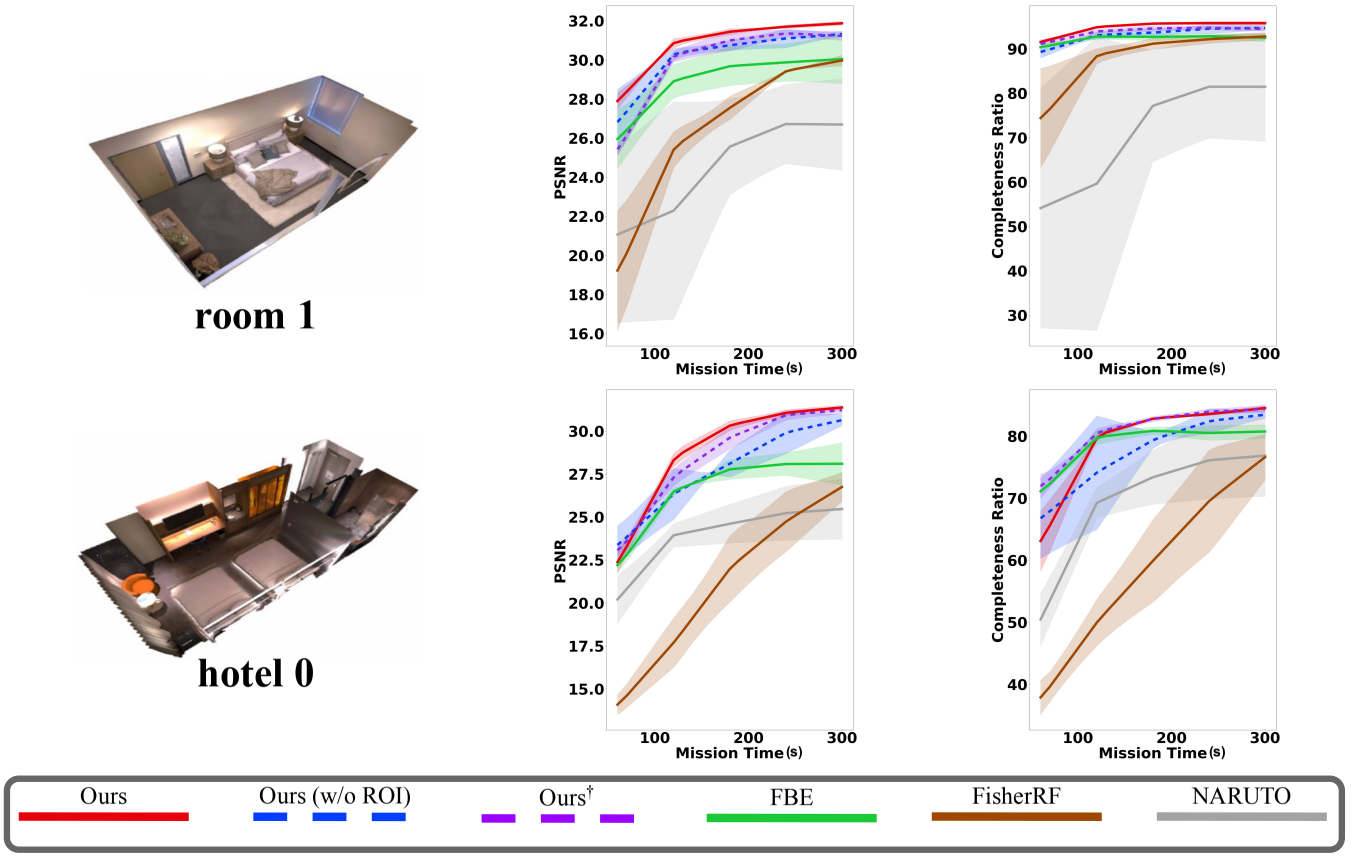


Figure 6.4: We report the reconstruction performance evaluated in rendering and mesh quality over online mission time. Our ActiveGS outperforms baselines in all test scenes. Our view planner considers unexplored regions for exploration, while exploiting low-confidence Gaussian primitives for further inspection. Compared to GS-based approaches, our approach proposes explicit confidence modeling of Gaussian primitives, enabling targeted candidate viewpoint generation and fast viewpoint evaluation. Our approaches demonstrate a large performance gain over the state-of-the-art NeRF-based approach, *NARUTO*, motivating the use of GS in active scene reconstruction.

We report the reconstruction performance over total mission time, defined as the summation of mapping time, planning time, and action time, assuming a constant robot velocity of 1 m/s. The reconstruction performance is evaluated on both rendering and mesh quality. For the rendering evaluation, we capture ground-truth RGB images from 1000 uniformly distributed test viewpoints in the scene's free space. We report PSNR [107] of RGB images rendered from our GS map at test viewpoints as the rendering quality metric. For mesh evaluation, we run TSDF fusion [114] on depth images rendered at training viewpoints and extract the scene mesh using marching cubes [99]. We use the completeness ratio [38] as the mesh quality metric with a distance threshold of 2 cm.

We consider the following methods:

- *Ours*: our full ActiveGS pipeline utilizing both exploration and exploitation utility measures. We consider regions-of-interest-based sampling to achieve targeted candidate viewpoint generation as described in Section 6.1.5;
- *Ours (w/o ROI)*: a variant of our ActiveGS that leverages only local random sampling, with $N_{\text{ROI}} = 0$;
- *Ours*†: a variant of our ActiveGS with an alternative confidence formulation, assigning higher confidence to Gaussian primitives with more visible viewpoints, without considering their spatial distribution;
- *FBE* [202]: a frontier-based exploration method that solely focuses on covering unexplored regions, without accounting for the quality of the GS map. We use the collected RGB-D measurements to update the GS map, similar to our approach;
- *FisherRF* [70]: a GS-based active scene reconstruction approach using only frontier voxels for regions-of-interest-based candidate viewpoint generation and Fisher information for viewpoint evaluation. We replace its 3D GS map with our 2D GS;
- *NARUTO* [38]: a state-of-the-art NeRF-based active scene reconstruction pipeline.

We run 5 trials for all methods across 8 test scenes. We set the maximum mission time to 300 s and evaluate reconstruction performance every 60 s. We report the mean and standard deviation for PSNR and completeness ratio.

We present the results of simulation experiments in Figure 6.4. Our approach achieves the best performance in both rendering and mesh quality across all test scenes, supporting our first claim that it outperforms state-of-the-art NeRF and GS-based methods. The NeRF-based active scene reconstruction approach,

Figure 6.5: Visual comparison of reconstruction results using different approaches. We show RGB rendering and surface meshes for two scenes, with red circles highlighting areas of low-quality reconstruction from baseline approaches. Our ActiveGS considers both unexplored regions in voxel map and confidence value of GS map to enable targeted view planning, achieving complete and high-fidelity scene reconstruction.

NARUTO, exhibits a significant performance gap compared to our approach, particularly in RGB rendering. This disparity arises because NeRF-based methods often compromise model capacity for faster map updates, limiting their representation quality in scene-level reconstruction. *FisherRF* evaluates viewpoint utility by calculating the Fisher information in the parameters of the Gaussian primitives within its field of view. This requires computationally expensive gradient calculation for all candidate and training viewpoints, leading to prolonged planning times and incomplete reconstruction under limited mission time. Additionally, since Fisher information is conditioned on the candidate viewpoint, the viewpoint must be selected before its utility can be evaluated, preventing direct viewpoint sampling informed by Fisher information. In contrast, our approach models the confidence of each Gaussian primitive, enabling fast feed-forward confidence rendering for viewpoint evaluation and identification of low-confidence surfaces for targeted candidate viewpoint generation, significantly enhancing reconstruction quality and efficiency. *FBE* focuses solely on exploration and ignores surface reconstruction quality, limiting its performance, while our approach balances exploration and exploitation by accounting for both unexplored regions and low-confidence Gaussian primitives.

The ablation study comparing *Ours* and *Ours (w/o ROI)* demonstrates the benefits of regions-of-interest-based sampling for targeted inspection, reflected by higher means and smaller standard deviations in both evaluation metrics. Our confidence formulation also outperforms the variant in *Ours*† by considering viewpoint distribution. These results confirm that our confidence modeling is effective in achieving efficient and high-fidelity active scene reconstruction, validating our second claim. We visually compare the reconstruction results in Figure 6.5.

6.2.3 Real-World Experiments

We demonstrate the applicability of our approach in a real-world experiment using a UAV equipped with an Intel RealSense 455 RGB-D camera to reconstruct a scene of size $6\,\mathrm{m} \times 6\,\mathrm{m} \times 3\,\mathrm{m}$. Unlike simulation experiments, we do not account for the pitch angle of viewpoints in this experiment due to control limitations. The UAV pose is tracked by an OptiTrack motion capture system. Given the limited onboard resources, we run ActiveGS on our desktop PC, where it receives RGB-D and pose data from the UAV for map updates and sends planned collision-free waypoints to guide the UAV. All communication is handled via ROS [138].

Our real-world experiments indicate that our approach is effective for actively reconstructing unknown scenes by considering both unexplored regions in the voxel map and under-reconstructed surfaces in the GS map. We show the experimental setup and the reconstructed GS map in Figure 6.6.

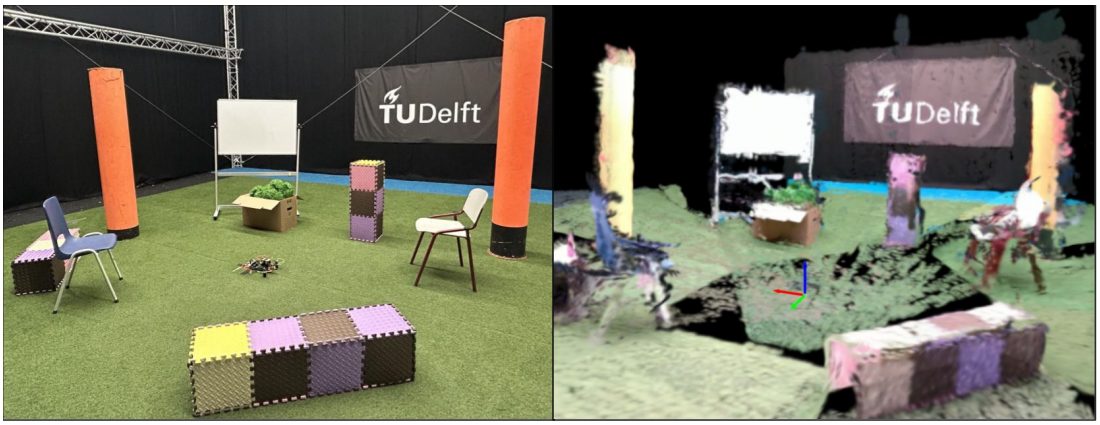


Figure 6.6: Our real-world experiments using a UAV equipped with an RGB-D camera. We show the experimental setup (left) and the RGB rendering from our GS map (right).

6.3 Related Work

Our work uses Gaussian splatting as the primary map representation for active scene reconstruction. In this section, we introduce state-of-the-art high-fidelity map representations, focusing on GS and active scene reconstruction methods.

6.3.1 Gaussian Splatting as Map Representation

Conventional map representations such as voxel grids [59], meshes [6, 198], and point clouds [210] often only capture coarse scene structures, struggling to provide fine-grained geometric and textural information crucial for many robotics applications [24]. Recent advances in implicit neural representations, such as NeRFs [107, 110], show promising results in high-fidelity scene reconstruction by modeling scene attributes continuously. Although they achieve impressive reconstruction results, NeRFs require dense sampling along rays for view synthesis, a computationally intensive process that limits their online applicability, as we discussed before in our previous approaches in Chapter 4 and Chapter 5.

3D GS [78] offers an efficient alternative for high-fidelity scene reconstruction by combining explicit map structures with volume rendering. Unlike NeRFs, 3D GS stores scene information using explicit 3D Gaussian primitives, eliminating the need for inefficient dense sampling during volume rendering. This explicit nature also makes it well-suited for online incremental mapping, which requires frequent measurement fusion and scene attributes modification. Follow-up works further enhance geometric quality by regularizing 3D GS training [47] or directly adopting 2D GS for improved surface alignment [29, 61]. 2D GS collapses the 3D primitive volume into 2D oriented planar Gaussian primitives, enabling more accurate depth estimation and allowing the integration of normal information

into the optimization process. Motivated by its strong performance, we utilize 2D GS, specifically Gaussian surfel [29], as our GS map representation for active scene reconstruction.

6.3.2 Active Scene Reconstruction

Active scene reconstruction using autonomous mobile robots is an area of active research [24]. Given an unknown scene, the goal is to explore and map the scene by actively planning the robot's next viewpoints for effective measurement acquisition. Traditional active scene reconstructions utilize map representations such as voxel maps [11,64,80,152,219], meshes [160,161], or point clouds [45,210]. These approaches primarily focus on fully covering the unknown space, rather than preserving fine-grained geometric and textural details of the scene. However, high-fidelity scene reconstruction is crucial for downstream robotic tasks that rely on accurate map information.

To address this, recent research explores implicit neural representations for active reconstruction applications. In an object-centric setup, several methods incorporate uncertainty estimation into NeRFs [124,139,203] and use this information to select next best viewpoints. For scene-level reconstruction, Yan *et al.* [204] and Kuang *et al.* [85] investigate the loss landscape of implicit neural representations during training to identify under-reconstructed areas. NARUTO [38] learns an uncertainty grid map alongside a hybrid neural scene representation, guiding measurement acquisition in uncertain regions. These implicit neural representations often face challenges such as inefficient map updates and catastrophic forgetting during incremental mapping.

Several works propose GS-based active scene reconstruction approaches. GS-Planner [71] and HGS-planner [201] incorporate unknown voxels into the GS rendering pipeline and detect unseen regions for exploration. Li *et al.* [92] use a Voronoi graph to extract a traversable topological map from the GS representation for path planning. The approach is designed for a 2D planning space, reducing its effectiveness in cluttered environments. FisherRF [70] evaluates the information content of a novel view by measuring the Fisher information value in the GS parameters. This procedure requires computationally expensive gradient calculations at each previously visited and candidate viewpoint, making view planning inefficient for online missions. We build upon the idea of using GS for active scene reconstruction, while introducing a key innovation: we explicitly model the confidence of each Gaussian primitive, enabling viewpoint sampling around low-confidence Gaussian primitives for targeted inspection and fast feed-forward confidence rendering for efficient viewpoint evaluation.

6.4 Conclusion

In this chapter, we propose ActiveGS, a GS-based active scene reconstruction approach. Our approach employs a hybrid map representation that combines the high-fidelity scene reconstruction capabilities of the GS map with the spatial modeling strengths of the voxel map. We present an effective method for confidence modeling of Gaussian primitives, enabling targeted viewpoint generation and informative viewpoint evaluation. Our view planning strategy leverages the confidence information of Gaussian primitives to inspect under-reconstructed areas, while also considering unexplored regions in the voxel map for exploration.

We conduct planning experiments in various indoor scenes in a simulator. Experimental results demonstrate that ActiveGS outperforms baseline approaches in both rendering and mesh quality, compared to state-of-the-art NeRF-based and GS-based active scene reconstruction approaches. We further validate our approach in real-world experiments using a UAV equipped with an RGB-D camera, demonstrating its applicability for active scene reconstruction.

Chapter 7

Conclusion

Autonomous robots must perceive and understand their environment to perform tasks successfully. A core aspect of this robot perception capability is to actively explore its surroundings and acquire informative sensor measurements. In contrast to passive perception, which follows predefined path patterns, fixed heuristics, or fully relies on external supervision, active perception leverages view planning to enable autonomous decision-making regarding where to gather measurements based on the robot's current understanding of the environment, allowing the robot to focus on acquiring the most valuable information for the task at hand. This is especially important when deploying robots in unknown environments under mission constraints, where no prior knowledge exists and the perception strategy must be optimized online. In such cases, effective view planning can significantly improve performance in tasks such as localization, object recognition, and mapping.

This thesis focuses on the problem of autonomous mapping in unknown environments and investigates the integration of active perception strategies with robot mapping systems. Our goal is to enable robots to construct accurate spatial representations of their surroundings by actively and intelligently collecting sensor measurements during missions. Although prior work has explored active perception for robot mapping, many existing approaches do not focus on preserving fine-grained environmental details, which bottlenecks their application in tasks requiring high-fidelity scene modeling. This limitation is largely caused by the reliance on conventional map representations in these approaches, often resulting in loss of geometric and textural information during the mapping process, due to their rigid and discrete nature. Our work is motivated by the challenge of actively building high-fidelity map representations that can capture scene details of an initially unknown environment.

To address this challenge, we propose novel approaches that leverage learning-based mapping techniques capable of modeling the environment in a continuous

manner. Such techniques allow for the generation of more accurate map representations, enabling robot mapping systems to retain fine-grained geometric and textural information that traditional methods fail to capture. The primary contribution of this thesis is the development of active perception approaches that integrate different learning-based mapping techniques to enable autonomous and high-fidelity robot mapping. Central to these approaches is the adaptation of map representations and the design of utility formulations that assess the expected usefulness of measurements taken at candidate viewpoints with respect to specific mapping objectives, enabling the integration of active perception strategies. To verify the effectiveness of our approaches, we conduct extensive evaluations in both simulated and real-world scenarios, demonstrating consistent improvements in mapping quality and computational efficiency over baselines.

7.1 Short Summary of the Key Contributions

This thesis presents four distinct active perception approaches for robot mapping, each leveraging different learning-based map representations capable of modeling scenes continuously to achieve high-fidelity reconstructions. We address our research question of effective integration of active perception strategies with learning-based mapping techniques from varying perspectives. Our solutions are based on the characteristics of the employed mapping methods and the specific mapping objectives. Below, we summarize the key contributions of each chapter and highlight the new capabilities enabled by this work, which were not possible at the beginning of this PhD research.

In Chapter 3, we explore GPs to model the spatial distribution of environmental properties, e.g., temperature distribution. For online incremental mapping, we initialize a spatially correlated grid map with a GP prior and perform sequential Bayesian fusion to incorporate new measurements. Leveraging the natural uncertainty modeling provided by GPs, we develop an active perception strategy in which the robot selects viewpoints that maximize uncertainty reduction in regions of interest through forward simulation. A significant contribution of this chapter is the introduction of an integral kernel for GPs, enabling the maintenance of an adaptive-resolution grid map in a computationally efficient and theoretically sound manner, a capability that was not previously available in GP fusion methods. This adaptive strategy retains the probabilistic nature of GPs while reducing resolution in less interesting areas, leading to lower memory usage and faster inference. These properties make it possible to perform efficient view planning in GP-based map representations during online missions. We demonstrate the effectiveness of this approach in a 2D temperature field mapping scenario using a UAV. We believe that this work could benefit scalar field mapping tasks,

such as weed density and soil moisture mapping in agricultural applications.

In Chapter 4, we address the challenge of photorealistic object modeling using an image-based neural rendering technique in the context of active perception. Our approach trains a neural network to synthesize photorealistic views from posed RGB reference images and aims to actively collect these reference images in an unknown scene to enhance the network's rendering performance. The key innovation here is probabilistically modeling the color rendering process in our network, providing uncertainty estimates based on the predicted variance of color rendering at each pixel. This uncertainty highlights areas where the network is less confident in the contents it renders, given the current reference images. This is one of the first approaches that provides uncertainty modeling in image-based neural rendering, and we further use this uncertainty to direct the robot to acquire new measurements at viewpoints with the highest predicted uncertainty to enhance its understanding of the scene. The collected images and the rendering network together form the internal map, enabling scene information retrieval from novel viewpoints. This introduces a novel NBV planning paradigm that allows the robot to gather informative measurements without explicitly updating a map during the mission, which could become costly, especially when using implicit neural representations or operating in large-scale environments.

In Chapter 5, we introduce a novel semantic-targeted active perception approach for robot mapping. Our goal is to selectively reconstruct task-relevant object classes in an unknown environment, while deprioritizing irrelevant areas. We combine NeRFs with semantic information as the map representation, and derive uncertainty estimates from the density distribution of NeRFs to identify regions of high geometric ambiguity. By rendering dense semantic and uncertainty views from novel viewpoints, our view planning method selects the most informative viewpoints for collecting new measurements, improving the reconstruction of target objects. This targeted view planning strategy is crucial in task-driven scenarios, where the robot must focus its mission budgets on important objects. For example, in robot manipulation, our approach enables the robot to efficiently reconstruct the objects of interest, providing important information for subsequent manipulation actions.

Different from the map representations in the previous two chapters that require computationally inefficient volume rendering, we utilize a more efficient learning-based map representation in Chapter 6. We propose one of the first active perception approaches for scene-level robot mapping based on GS. Our approach combines GS with a coarse voxel map to leverage the strengths of both representations: high-fidelity scene reconstruction with GS and spatial modeling with voxels. The core in this approach is an effective confidence modeling technique that assigns confidence values to each primitive in the GS map based on

the viewpoint distribution, helping to identify under-reconstructed areas. Additionally, we use voxel map information to target unexplored regions and assist with collision-free path planning. By actively collecting measurements in both under-reconstructed and unexplored areas, our approach achieves superior GS reconstruction in indoor scenarios.

Overall, the contributions presented in this thesis support our motivation to integrate active perception with learning-based map representations and highlight the potential of this integration to enable high-fidelity autonomous robot mapping in unknown environments. Furthermore, the different approaches introduced in this thesis offer a glimpse into the rapid evolution of learning-based map representations within a short period, inspiring future research to further advance their capabilities.

7.2 Future Work

Besides the research conducted in this thesis, we also identify several promising areas for future work that could further enhance the mapping quality and efficiency in autonomous robot mapping tasks. Specifically, we outline here three main areas for future exploration: (1) leveraging prior information from pretrained models; (2) extending active perception for robot mapping to active simultaneous localization and mapping (SLAM) systems; and (3) scaling the approach to multi-robot systems.

The first area involves exploiting prior information contained in pretrained models for robot mapping tasks. For instance, we could utilize diffusion models [56] to generate 3D contents from 2D images [81, 93, 94, 95], which can be directly integrated into map representations to complete missing information or provide a proxy to ground truth to guide the view planning toward incomplete areas. Although the application of diffusion models in active perception for robot mapping is still largely limited by its computational cost, combining the generative capabilities with the active perception strategies could lead to more efficient and robust mapping. Another promising direction in this domain is to leverage recent foundation models for 3D reconstruction [91, 191, 195, 196]. By training on large-scale 3D datasets, these models provide strong geometric priors conditioned on a set of unposed RGB images or video sequences. This can be particularly useful for RGB-based mapping tasks, where the robot can utilize these priors as a starting point, and combined with active perception strategies to allow the robot to focus on refining the map in areas where the prior is uncertain or incomplete.

In this thesis, we primarily focused on active perception for robot mapping, assuming that the robot is already localized in the environment. However, in many real-world scenarios, the robot needs to perform SLAM to build a map of the

environment while also keeping track of its own position. From this perspective, active perception for robot mapping can be seen as a subproblem of active SLAM, where the goal shifts toward optimizing mapping objectives without the additional complexity of simultaneous localization. Therefore, the second area for future work would be extending the active perception for robot mapping system to active SLAM systems. To achieve active SLAM, we need to integrate active perception strategies with both mapping and localization objectives, ensuring that the robot can effectively build a map of the environment while maintaining accurate localization. Some recent works have explored the integration of active perception with SLAM systems by considering both localization and mapping uncertainties for utility formulation [5, 15, 88]. Nevertheless, active SLAM with learning-based map representations remains a largely open research challenge.

The third area for future work involves extending the active perception for robot mapping to multi-robot systems [2, 166, 187, 200]. While this thesis primarily focused on a single-robot setup, many real-world scenarios, especially for mapping large-scale environments, could benefit significantly from collaborative multi-robot systems. In such systems, multiple robots must coordinate their perception strategies to efficiently explore and reconstruct the environment. This entails developing algorithms for map merging, inter-robot information sharing, and coordinated decision-making, all while maintaining consistency and efficiency throughout the mapping process.

Bibliography

[1] A. Asgharivaskasi, S. Koga, and N. Atanasov. Active Mapping via Gradient Ascent Optimization of Shannon Mutual Information over Continuous SE(3) Trajectories. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.

[2] N. Atanasov, J. Le Ny, K. Daniilidis, and G. Pappas. Decentralized Active Information Acquisition: Theory and Application to Multi-Robot SLAM. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2015.

[3] C. Bai, T. Xiao, Y. Chen, H. Wang, F. Zhang, and X. Gao. Faster-LIO: Lightweight Tightly Coupled Lidar-Inertial Odometry Using Parallel Sparse Incremental Voxels. *IEEE Robotics and Automation Letters (RA-L)*, 7(2):4861–4868, 2022.

[4] R. Bajcsy, Y. Aloimonos, and J. Tsotsos. Revisiting Active Perception. *Autonomous Robots*, 42:177–196, 2018.

[5] A. Batinovic, T. Petrovic, A. Ivanovic, F. Petric, and S. Bogdan. A Multi-Resolution Frontier-Based Planner for Autonomous 3D Exploration. *IEEE Robotics and Automation Letters (RA-L)*, 6(3):4528–4535, 2021.

[6] J. Behley and C. Stachniss. Efficient Surfel-Based SLAM Using 3D Laser Range Data in Urban Environments. In *Proc. of Robotics: Science and Systems (RSS)*, 2018.

[7] M. Besselmann, L. Puck, L. Steffen, A. Rönnau, and R. Dillmann. VDB-Mapping: A High Resolution and Real-Time Capable 3D Mapping Framework for Versatile Mobile Robots. In *Proc. of the Intl. Conf. on Automation Science and Engineering (CASE)*, 2021.

[8] Y. Bhalgat, I. Laina, J. Henriques, A. Zisserman, and A. Vedaldi. Contrastive Lift: 3D Object Instance Segmentation by Slow-Fast Contrastive Fusion. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2023.

[9] P. Biber and T. Duckett. Dynamic Maps for Long-Term Operation of Mobile Service Robots. In *Proc. of Robotics: Science and Systems (RSS)*, 2005.

[10] A. Bircher, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel, and R. Siegwart. Structural Inspection Path Planning via Iterative Viewpoint Resampling with Application to Aerial Robotics. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2015.

[11] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart. Receding Horizon "Next-Best-View" Planner for 3D Exploration. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2016.

[12] A. Bircher, M. Kamel, K. Alexis, H. Oleynikova, and R. Siegwart. Receding Horizon Path Planning for 3D Exploration and Surface Inspection. *Autonomous Robots*, 42(2):291–306, 2018.

[13] F. Blöchliger, M. Fehr, M. Dymczyk, T. Schneider, and R. Siegwart. Topomap: Topological Mapping and Navigation Based on Visual SLAM Maps. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2018.

[14] N. Blodow, L. Goron, Z. Marton, D. Pangercic, T. Rühr, M. Tenorth, and M. Beetz. Autonomous Semantic Mapping for Robots Performing Everyday Manipulation Tasks in Kitchen Environments. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2011.

[15] E. Bonetto, P. Goldschmid, M. Pabst, M. Black, and A. Ahmad. iRotate: Active Visual SLAM for Omnidirectional Robots. *Journal on Robotics and Autonomous Systems (RAS)*, 154:104102, 2022.

[16] W. Burgard, M. Moors, C. Stachniss, and F. Schneider. Coordinated Multi-Robot Exploration. *IEEE Trans. on Robotics (TRO)*, 21(3):376–378, 2005.

[17] A. Burusa, J. Scholten, D. Rincon, X. Wang, E. Van Henten, and G. Kootstra. Efficient Search and Detection of Relevant Plant Parts Using Semantics-Aware Active Vision. *Biosystems Engineering*, 248:1–14, 2024.

[18] A. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu. ShapeNet: An Information-Rich 3D Model Repository. *arXiv preprint*, arXiv:1512.03012, 2015.

[19] D. Chaplot, E. Parisotto, and R. Salakhutdinov. Active Neural Localization. In *Proc. of the Intl. Conf. on Learning Representations (ICLR)*, 2018.

[20] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su. TensoRF: Tensorial Radiance Fields. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2022.

[21] G. Chen and W. Wang. A Survey on 3D Gaussian Splatting. *arXiv preprint*, arXiv:2401.03890, 2024.

[22] J. Chen and S. Shen. Improving Octree-Based Occupancy Maps Using Environment Sparsity with Application to Aerial Robot Navigation. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[23] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(4):834–848, 2018.

[24] S. Chen, Y. Li, and N. Kwok. Active Vision in Robotic Systems: A Survey of Recent Developments. *Intl. Journal of Robotics Research (IJRR)*, 30(11):1343–1377, 2011.

[25] W. Chen, R. Khardon, and L. Liu. AK: Attentive Kernel for Information Gathering. In *Proc. of Robotics: Science and Systems (RSS)*, 2022.

[26] Y. Chen, W. Shuai, and X. Chen. A Probabilistic, Variable-Resolution and Effective Quadtree Representation for Mapping of Large Environments. In *Proc. of the Intl. Conf. on Advanced Robotics (ICAR)*, 2015.

[27] T. Cieslewski, E. Kaufmann, and D. Scaramuzza. Rapid Exploration with Multi-Rotors: A Frontier Selection Method for High Speed Flight. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[28] C. Collander, W. Beksi, and M. Huber. Learning the Next Best View for 3D Point Clouds via Topological Features. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2021.

[29] P. Dai, J. Xu, W. Xie, X. Liu, H. Wang, and W. Xu. High-Quality Surface Reconstruction Using Gaussian Surfels. In *Proc. of the Intl. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 2024.

[30] J. Delmerico, S. Isler, R. Sabzevari, and D. Scaramuzza. A Comparison of Volumetric Information Gain Metrics for Active 3D Object Reconstruction. *Autonomous Robots*, 42:197–208, 2018.

[31] H. Dhami, V. Sharma, and P. Tokekar. Pred-NBV: Prediction-Guided Next-Best-View Planning for 3D Object Reconstruction. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2023.

[32] W. Ding, N. Majcherczyk, M. Deshpande, X. Qi, D. Zhao, R. Madhivanan, and A. Sen. Learning to View: Decision Transformers for Active Object Detection. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2023.

[33] W. Dong, J. Shi, W. Tang, X. Wang, and H. Zha. An Efficient Volumetric Mesh Representation for Real-Time Scene Reconstruction Using Spatial Hashing. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2018.

[34] R. Dube, M. Gollub, H. Sommer, I. Gilitschenski, R. Siegwart, C.Cadena, and J. Nieto. Incremental Segment-Based Localization in 3D Point Clouds. *IEEE Robotics and Automation Letters (RA-L)*, 3(2):1832–1839, 2018.

[35] E. Einhorn, C. Schröter, and H. Gross. Finding the Adequate Resolution for Grid Mapping - Cell Sizes Locally Adapting On-the-Fly. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2011.

[36] A. Elfes. Using Occupancy Grids for Mobile Robot Perception and Navigation. *Computer*, 22(6):46–57, 1989.

[37] F. Engelmann, M. Bokeloh, A. Fathi, B. Leibe, and M. Nießner. 3D-MPA: Multi-Proposal Aggregation for 3D Semantic Instance Segmentation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[38] Z. Feng, H. Zhan, Z. Chen, Q. Yan, X. Xu, C. Cai, B. Li, Q. Zhu, and Y. Xu. NARUTO: Neural Active Reconstruction from Uncertain Target Observations. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[39] L. Fermin-Leon, J. Neira, and J. Castellanos. Incremental Contour-Based Topological Segmentation for Robot Exploration. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[40] D. Fox, W. Burgard, and S. Thrun. Active Markov Localization for Mobile Robots. *Journal on Robotics and Autonomous Systems (RAS)*, 25(3–4):195–207, 1998.

[41] B. Frank, R. Schmedding, C. Stachniss, M. Teschner, and W. Burgard. Learning the Elasticity Parameters of Deformable Objects with a Manipulation Robot. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2010.

[42] B. Frank, C. Stachniss, N. Abdo, and W. Burgard. Efficient Motion Planning for Manipulation Robots in Environments with Deformable Objects. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2011.

[43] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa. Plenoxels: Radiance Fields without Neural Wetworks. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[44] N. Funk, J. Tarrio, S. Papatheodorou, M. Popović, P. Alcantarilla, and S. Leutenegger. Multi-Resolution 3D Mapping with Explicit Free Space Representation for Fast and Accurate Mobile Robot Motion Planning. *IEEE Robotics and Automation Letters (RA-L)*, 6(2):3553–3560, 2021.

[45] Y. Gao, Y. Wang, X. Zhong, T. Yang, M. Wang, Z. Xu, Y. Wang, Y. Lin, C. Xu, and F. Gao. Meeting-Merging-Mission: A Multi-Robot Coordinate Framework for Large-Scale Communication-Limited Exploration. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.

[46] D. Gregorio and L. Stefano. SkiMap: An Efficient Mapping Framework for Robot Navigation. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[47] A. Guédon and V. Lepetit. Sugar: Surface-Aligned Gaussian Splatting for Efficient 3D Mesh Reconstruction and High-Quality Mesh Rendering. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[48] L. Han, F. Gao, B. Zhou, and S. Shen. Fiesta: Fast Incremental Euclidean Distance Fields for Online Motion Planning of Aerial Robots. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.

[49] M. Hanlon, B. Sun, M. Pollefeys, and H. Blum. Active Visual Localization for Multi-Agent Collaboration: A Data-Driven Approach. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2024.

[50] Z. Hao, D. Romero, T. Lin, and M. Liu. Meshtron: High-Fidelity, Artist-Like 3D Mesh Generation at Scale. *arXiv preprint*, arXiv:2412.09548, 2024.

[51] G. Hardouin, J. Moras, F. Morbidi, J. Marzat, and E. Mouaddib. Next-Best-View Planning for Surface Reconstruction of Large-Scale 3D Environments with Multiple UAVs. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

[52] P. Hart, N. Nilsson, and B. Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Trans. on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[53] S. He, C. Hsu, D. Ong, Y. Shao, and P. Chaudhari. Active Perception Using Neural Radiance Fields. In *Proc. of the IEEE American Control Conf. (ACC)*, 2024.

[54] G. Hitz, E. Galceran, M. Garneau, F. Pomerleau, and R. Siegwart. Adaptive Continuous-Space Informative Path Planning for Online Environmental Monitoring. *Journal of Field Robotics (JFR)*, 34(8):1427–1449, 2017.

[55] G. Hitz, A. Gotovos, F. Pomerleau, M. Garneau, C. Pradalier, A. Krause, and R. Siegwart. Fully Autonomous Focused Exploration for Robotic Environmental Monitoring. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2014.

[56] J. Ho, A. Jain, and P. Abbeel. Denoising Diffusion Probabilistic Models. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2020.

[57] S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780, 1997.

[58] G. Hollinger and G. Sukhatme. Sampling-Based Robotic Information Gathering Algorithms. *Intl. Journal of Robotics Research (IJRR)*, 33(9):1271–1287, 2014.

[59] A. Hornung, K. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard. OctoMap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees. *Autonomous Robots*, 34(3):189–206, 2013.

[60] H. Hu, S. Pan, L. Jin, M. Popović, and M. Bennewitz. Active Implicit Reconstruction Using One-Shot View Planning. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2024.

[61] B. Huang, Z. Yu, A. Chen, A. Geiger, and S. Gao. 2D Gaussian Splatting for Geometrically Accurate Radiance Fields. In *Proc. of the Intl. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 2024.

[62] J. Hurtado and A. Valada. Semantic Scene Segmentation for Robotics. In *Deep Learning for Robot Perception and Cognition*. 2022.

[63] E. Ilg, Ö. Çiçek, S. Galesso, A. Klein, O. Makansi, F. Hutter, and T. Brox. Uncertainty Estimates and Multi-Hypotheses Networks for Optical Flow. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2018.

[64] S. Isler, R. Sabzevari, J. Delmerico, and D. Scaramuzza. An Information Gain Formulation for Active Volumetric 3D Reconstruction. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2016.

[65] M. Jadidi, J. Miro, and G. Dissanayake. Warped Gaussian Processes Occupancy Mapping with Uncertain Inputs. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[66] M. Jadidi, J. Miro, and G. Dissanayake. Gaussian Processes Autonomous Mapping and Exploration for Range-Sensing Mobile Robots. *Autonomous Robots*, 42(2):273–290, 2018.

[67] A. Jain, M. Tancik, and P. Abbeel. Putting NeRF on a Diet: Semantically Consistent Few-Shot View Synthesis. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[68] R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanæs. Large Scale Multi-View Stereopsis Evaluation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[69] C. Jiang, H. Zhang, P. Liu, Z. Yu, H. Cheng, B. Zhou, and S. Shen. H_2-Mapping: Real-Time Dense Mapping Using Hierarchical Hybrid Representation. *IEEE Robotics and Automation Letters (RA-L)*, 8(10):6787–6794, 2023.

[70] W. Jiang, B. Lei, and K. Daniilidis. FisherRF: Active View Selection and Mapping with Radiance Fields Using Fisher Information. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2024.

[71] R. Jin, Y. Gao, H. Lu, and F. Gao. GS-Planner: A Gaussian-Splatting-Based Planning Framework for Active High-Fidelity Reconstruction. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2024.

[72] M. Kaba, M. Uzunbas, and S. Lim. A Reinforcement Learning Approach to the View Planning Problem. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[73] O. Kähler, V. Prisacariu, J. Valentin, and D. Murray. Hierarchical Voxel Block Hashing for Efficient Integration of Depth Images. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2016.

[74] J. Kajiya and B. Von Herzen. Ray Tracing Volume Densities. *Proc. of the Intl. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1984.

[75] N. Keetha, J. Karhade, K. Jatavallabhula, G. Yang, S. Scherer, D. Ramanan, and J. Luiten. SplaTAM: Splat, Track & Map 3D Gaussians for Dense RGB-D SLAM. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[76] S. Kelly, A. Riccardi, E. Marks, F. Magistri, T. Guadagnino, M. Chli, and C. Stachniss. Target-Aware Implicit Mapping for Agricultural Crop Inspection. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2023.

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

[78] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. on Graphics (TOG)*, 42(4):139–1, 2023.

[79] S. Kim and J. Kim. Recursive Bayesian Updates for Occupancy Mapping and Surface Reconstruction. In *Proc. of the Australasian Conf. on Robotics and Automation (ACRA)*, 2014.

[80] Y. Kompis, L. Bartolomei, R. Mascaro, L. Teixeira, and M. Chli. Informed Sampling Exploration Path Planner for 3D Reconstruction of Large Scenes. *IEEE Robotics and Automation Letters (RA-L)*, 6(4):7893–7900, 2021.

[81] X. Kong, S. Liu, X. Lyu, M. Taher, X. Qi, and A. Davison. EscherNet: A Generative Model for Scalable View Synthesis. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[82] G. Kraetzschmar, G. Gassull, and K. Uhl. Probabilistic Quadtrees for Variable-Resolution Mapping of Large Environments. *IFAC/EURON Symposium on Intelligent Autonomous Vehicles*, 37(8):675–680, 2004.

[83] A. Krause, A. Singh, and C. Guestrin. Near-Optimal Sensor Placements in Gaussian Processes: Theory, Efficient Algorithms and Empirical Studies. *Journal on Machine Learning Research (JMLR)*, 9:235–284, 2008.

[84] L. Kreuzberg, I. Zulfikar, S. Mahadevan, F. Engelmann, and B. Leibe. 4D-StOP: Panoptic Segmentation of 4D LiDAR Using Spatio-Temporal Object Proposal Generation and Aggregation. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2022.

[85] Z. Kuang, Z. Yan, H. Zhao, G. Zhou, and H. Zha. Active Neural Mapping at Scale. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2024.

[86] C. Landgraf, B. Meese, M. Pabst, G. Martius, and M. Huber. A Reinforcement Learning Approach to View Planning for Automated Inspection Tasks. *Sensors*, 21(6), 2021.

[87] S. LaValle. Rapidly-Exploring Random Trees: A New Tool for Path Planning. Technical report, Iowa State University, 1998.

[88] E. Lee, J. Choi, H. Lim, and H. Myung. REAL: Rapid Exploration with Active Loop-Closing toward Large-Scale 3D Mapping Using UAVs. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021.

[89] S. Lee, L. Chen, J. Wang, A. Liniger, S. Kumar, and F. Yu. Uncertainty Guided Policy for Active Robotic 3D Reconstruction Using Neural Radiance Fields. *IEEE Robotics and Automation Letters (RA-L)*, 7(4):12070–12077, 2022.

[90] C. Lehnert, D. Tsai, A. Eriksson, and C. McCool. 3D Move to See: Multi-Perspective Visual Servoing towards the Next Best View within Unstructured and Occluded Environments. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.

[91] V. Leroy, Y. Cabon, and J. Revaud. Grounding Image Matching in 3D with Mast3r. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2024.

[92] Y. Li, Z. Kuang, T. Li, G. Zhou, S. Zhang, and Z. Yan. ActiveSplat: High-Fidelity Scene Reconstruction through Active Gaussian Splatting. *IEEE Robotics and Automation Letters (RA-L)*, 10(8):8099–8106, 2025.

[93] M. Liu, R. Shi, L. Chen, Z. Zhang, C. Xu, X. Wei, H. Chen, C. Zeng, J. Gu, and H. Su. One-2-3-45++: Fast Single Image to 3D Objects with Consistent Multi-View Generation and 3D Diffusion. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[94] M. Liu, C. Xu, H. Jin, L. Chen, M. Varma, Z. Xu, and H. Su. One-2-3-45: Any Single Image to 3D Mesh in 45 Seconds without Per-Shape Optimization. *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 36, 2023.

[95] R. Liu, R. Wu, B. Van Hoorick, P. Tokmakov, S. Zakharov, and C. Vondrick. Zero-1-to-3: Zero-Shot One Image to 3D Object. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2023.

[96] I. Lluvia, E. Lazkano, and A. Ansuategi. Active Mapping and Robot Exploration: A Survey. *Sensors*, 21(7), 2021.

[97] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[98] A. Loquercio, M. Dymczyk, B. Zeisl, S. Lynen, R. Siegwart, and I. Gilitschenski. Efficient Descriptor Learning for Large Scale Localization. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[99] W. Lorensen and H. Cline. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. In *Proc. of the Intl. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1987.

[100] K. Ma, L. Liu, and G. Sukhatme. Informative Planning and Online Learning with Sparse Gaussian Processes. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[101] S. Manfreda, M. McCabe, P. Miller, R. Lucas, V. Pajuelo, G. Mallinis, E. Bendor, D. Helman, L. Estes, G. Ciraolo, J. Müllerová, F. Tauro, M. De Lima, J. De Lima, A. Maltese, F. Frances, K. Caylor, M. Kohv, M. Perks, G. Ruiz-Pérez, Z. Su, G. Vico, and B. Toth. On the Use of Unmanned Aerial Systems for Environmental Monitoring. *Remote Sensing*, 10(4), 2018.

[102] R. Martin-Brualla, N. Radwan, M. Sajjadi, J. Barron, A. Dosovitskiy, and D. Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[103] R. Mascaro and M. Chli. Scene Representations for Robotic Spatial Perception. *Annual Review of Control, Robotics, and Autonomous Systems*, 2024.

[104] H. Matsuki, R. Murai, P. Kelly, and A. Davison. Gaussian Splatting SLAM. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[105] J. McCormac, A. Handa, A. Davison, and S. Leutenegger. SemanticFusion: Dense 3D Semantic Mapping with Convolutional Neural Networks. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[106] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger. Occupancy Networks: Learning 3D Reconstruction in Function Space. In

Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), 2019.

[107] B. Mildenhall, P. Srinivasan, M. Tancik, J. Barron, R. Ramamoorthi, and R. Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2020.

[108] A. Milioto, I. Vizzo, J. Behley, and C. Stachniss. RangeNet++: Fast and Accurate LiDAR Semantic Segmentation. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2019.

[109] H. Moravec and A. Elfes. High Resolution Maps from Wide Angle Sonar. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 1985.

[110] T. Müller, A. Evans, C. Schied, and A. Keller. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *ACM Trans. on Graphics*, 41(4):102:1–102:15, 2022.

[111] R. Mur-Artal, J. Montiel, and J. Tardos. ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Trans. on Robotics (TRO)*, 31(5):1147–1163, 2015.

[112] K. Museth, J. Lait, J. Johanson, J. Budsberg, R. Henderson, M. Alden, P. Cucka, D. Hill, and A. Pearce. OpenVDB: An Open-Source Data Structure and Toolkit for High-Resolution Volumes. In *ACM SIGGRAPH Courses*. 2013.

[113] M. Naazare, F. Rosas, and D. Schulz. Online Next-Best-View Planner for 3D-Exploration and Inspection with a Mobile Manipulator Robot. *IEEE Robotics and Automation Letters (RA-L)*, 7(2):3779–3786, 2022.

[114] R. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon. KinectFusion: Real-Time Dense Surface Mapping and Tracking. In *Proc. of the Intl. Symp. on Mixed and Augmented Reality (ISMAR)*, 2011.

[115] M. Nießner, M. Zollhöfer, S. Izadi, and M. Stamminger. Real-Time 3D Reconstruction at Scale Using Voxel Hashing. In *Proc. of the SIGGRAPH Asia*, 2013.

[116] S. O'Callaghan and F. Ramos. Continuous Occupancy Mapping with Integral Kernels. In *Proc. of the Conf. on Advancements of Artificial Intelligence (AAAI)*, 2011.

[117] S. O'Callaghan and F. Ramos. Gaussian Process Occupancy Maps. *Intl. Journal of Robotics Research (IJRR)*, 31(1):42–62, 2012.

[118] M. Oechsle, S. Peng, and A. Geiger. Unisurf: Unifying Neural Implicit Surfaces and Radiance Fields for Multi-View Reconstruction. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[119] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto. Voxblox: Incremental 3D Euclidean Signed Distance Fields for On-Board MAV Planning. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[120] M. Oliver and R. Webster. Kriging: A Method of Interpolation for Geographical Information Systems. *International Journal of Geographical Information Systems*, 4(3):313–332, 1990.

[121] J. Ortiz, A. Clegg, J. Dong, E. Sucar, D. Novotny, M. Zollhöfer, and M. Mukadam. iSDF: Real-Time Neural Signed Distance Fields for Robot Perception. In *Proc. of Robotics: Science and Systems (RSS)*, 2022.

[122] E. Palazzolo and C. Stachniss. Effective Exploration for MAVs Based on the Expected Information Gain. *Drones*, 2(1), 2018.

[123] S. Pan, L. Jin, H. Hu, M. Popović, and M. Bennewitz. How Many Views Are Needed to Reconstruct an Unknown Object Using NeRF? In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2024.

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

[125] Y. Pan, Y. Kompis, L. Bartolomei, R. Mascaro, C. Stachniss, and M. Chli. Voxfield: Non-Projective Signed Distance Fields for Online Planning and 3D Reconstruction. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2022.

[126] C. Papachristos, S. Khattak, and K. Alexis. Uncertainty-Aware Receding Horizon Exploration and Mapping Using Aerial Robots. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[127] S. Papatheodorou, N. Funk, D. Tzoumanikas, C. Choi, B. Xu, and S. Leutenegger. Finding Things in the Unknown: Semantic Object-Centric Exploration with an MAV. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2023.

[128] J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove. DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[129] T. Patten, M. Zillich, R. Fitch, M. Vincze, and S. Sukkarieh. Viewpoint Evaluation for Online 3D Active Object Classification. *IEEE Robotics and Automation Letters (RA-L)*, 1(1):73–81, 2016.

[130] S. Peng, M. Niemeyer, L. Mescheder, M. Pollefeys, and A. Geiger. Convolutional Occupancy Networks. In *Proc. of the Europ. Conf. on Computer Vision (ECCV)*, 2020.

[131] E. Piazza, A. Romanoni, and M. Matteucci. Real-Time CPU-Based Large-Scale 3D Mesh Reconstruction. *IEEE Robotics and Automation Letters (RA-L)*, 3(3):1584–1591, 2018.

[132] R. Pito. A Solution to the Next Best View Problem for Automated Surface Acquisition. *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)*, 21(10):1016–1030, 1999.

[133] M. Popović, T. Vidal-Calleja, G. Hitz, J. Chung, I. Sa, R. Siegwart, and J. Nieto. An Informative Path Planning Framework for UAV-Based Terrain Monitoring. *Autonomous Robots*, 44(6):889–911, 2020.

[134] M. Popović, T. Vidal-Calleja, G. Hitz, I. Sa, R. Siegwart, and J. Nieto. Multiresolution Mapping and Informative Path Planning for UAV-Based Terrain Monitoring. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[135] A. Pumarola, E. Corona, G. Pons-Moll, and F. Moreno-Noguer. D-NeRF: Neural Radiance Fields for Dynamic Scenes. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[136] C. Qi, H. Su, K. Mo, and L. Guibas. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[137] J. Quenzel and S. Behnke. Real-Time Multi-Adaptive-Resolution-Surfel 6D LiDAR Odometry Using Continuous-Time Trajectory Optimization. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021.

[138] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Ng. ROS: An Open-Source Robot Operating System. In *Proc. of the ICRA Workshop on Open Source Software*, 2009.

[139] Y. Ran, J. Zeng, S. He, J. Chen, L. Li, Y. Chen, G. Lee, and Q. Ye. NeurAR: Neural Uncertainty for Autonomous 3D Reconstruction with Implicit Neural Representations. *IEEE Robotics and Automation Letters (RA-L)*, 8(2):1125–1132, 2023.

[140] C. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[141] S. Reece and S. Roberts. An Introduction to Gaussian Processes for the Kalman Filter Expert. In *Proc. of the Intl. Conf. on Information Fusion*, 2010.

[142] A. Reid, F. Ramos, and S. Sukkarieh. Bayesian Fusion for Multi-Modal Aerial Images. In *Proc. of Robotics: Science and Systems (RSS)*, 2013.

[143] P. Ren, Y. Xiao, X. Chang, P. Huang, Z. Li, B. Gupta, X. Chen, and X. Wang. A Survey of Deep Active Learning. *ACM Computing Surveys*, 54:1–40, 2021.

[144] B. Roessle, J. Barron, B. Mildenhall, P. Srinivasan, and M. Nießner. Dense Depth Priors for Neural Radiance Fields from Sparse Input Views. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[145] R. Rosu and S. Behnke. NeuralMVS: Bridging Multi-View Stereo and Novel View Synthesis. In *Proc. of the Intl. Joint Conf. on Neural Networks (IJCNN)*, 2022.

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

[147] L. Rudin and S. Osher. Total Variation Based Image Restoration with Free Local Constraints. In *Proc. of the IEEE Intl. Conf. on Image Processing (ICIP)*, 1994.

[148] S. Särkkä. Linear Operators and Stochastic Partial Differential Equations in Gaussian Process Regression. *Lecture Notes in Computer Science*, 6792(2):151–158, 2011.

[149] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, D. Parikh, and D. Batra. Habitat: A Platform for Embodied AI Research. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2019.

[150] L. Schmid, J. Delmerico, J. Schönberger, J. Nieto, M. Pollefeys, R. Siegwart, and C. 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 and Automation (ICRA)*, 2022.

[151] L. Schmid, C. Ni, Y. Zhong, R. Siegwart, and O. Andersson. Fast and Compute-Efficient Sampling-Based Local Exploration Planning via Distribution Learning. *IEEE Robotics and Automation Letters (RA-L)*, 7(3):7810–7817, 2022.

[152] L. Schmid, M. Pantic, R. Khanna, L. Ott, R. Siegwart, and J. 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.

[153] R. Senanayake and F. Ramos. Bayesian Hilbert Maps for Dynamic Continuous Occupancy Mapping. In *Proc. of the Conf. on Robot Learning (CoRL)*, 2017.

[154] J. Shen, A. Ruiz, A. Agudo, and F. Moreno-Noguer. Stochastic Neural Radiance Fields: Quantifying Uncertainty in Implicit 3D Representations. In *Proc. of the Intl. Conf. on 3D Vision (3DV)*, 2021.

[155] Y. Shen, A. Ng, and M. Seeger. Fast Gaussian Process Regression Using KD-Trees. In *Proc. of the Conf. Neural Information Processing Systems (NIPS)*, 2005.

[156] Y. Siddiqui, L. Porzi, S. Bulò, N. Müller, M. Nießner, A. Dai, and P. Kontschieder. Panoptic Lifting for 3D Scene Understanding with Neural Fields. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[157] V. Sitzmann, M. Zollhöfer, and G. Wetzstein. Scene Representation Networks: Continuous 3D-Structure-Aware Neural Scene Representations. In *Proc. of the Conf. on Neural Information Processing Systems (NeurIPS)*, 2019.

[158] E. Smith, M. Drozdzal, D. Nowrouzezahrai, D. Meger, and A. Romero-Soriano. Uncertainty-Driven Active Vision for Implicit Scene Reconstruction. *arXiv preprint*, arXiv:2210.00978, 2022.

[159] E. Snelson and Z. Ghahramani. Sparse Gaussian Processes Using Pseudo-Inputs. In *Proc. of the Conf. Neural Information Processing Systems (NIPS)*, 2006.

[160] S. Song and S. Jo. Online Inspection Path Planning for Autonomous 3D Modeling Using a Micro-Aerial Vehicle. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[161] S. Song and S. Jo. Surface-Based Exploration for Autonomous 3D Modeling. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2018.

[162] C. Spearman. The Proof and Measurement of Association between Two Things. *The American Journal of Psychology*, pages 441–471, 1904.

[163] F. Stache, J. Westheider, F. Magistri, M. Popović, and C. Stachniss. Adaptive Path Planning for UAV-Based Multi-Resolution Semantic Segmentation. In *Proc. of the Europ. Conf. on Mobile Robotics (ECMR)*, 2021.

[164] C. Stachniss, G. Grisetti, and W. Burgard. Information Gain-Based Exploration Using Rao-Blackwellized Particle Filters. In *Proc. of Robotics: Science and Systems (RSS)*, 2005.

[165] C. Stachniss, D. Hähnel, and W. Burgard. Exploration with Active Loop-Closing for FastSLAM. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2004.

[166] C. Stachniss, O. Martínez-Mozos, and W. Burgard. Speeding-Up Multi-Robot Exploration by Considering Semantic Place Information. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2006.

[167] C. Stachniss, C. Plagemann, and A. Lilienthal. Gas Distribution Modeling Using Sparse Gaussian Process Mixtures. *Autonomous Robots*, 26(2):187ff, 2009.

[168] C. Stachniss, C. Plagemann, A. Lilienthal, and W. Burgard. Gas Distribution Modeling Using Sparse Gaussian Process Mixture Models. In *Proc. of Robotics: Science and Systems (RSS)*, 2008.

[169] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. Engel, R. Mur-Artal, C. Ren, S. Verma, A. Clarkson, M. Yan, B. Budge, Y. Yan, X. Pan, J. Yon, Y. Zou, K. Leon, N. Carter, J. Briales, T. Gillingham,

E. Mueggler, L. Pesqueira, M. Savva, D. Batra, H. Strasdat, R. De Nardi, M. Goesele, S. Lovegrove, and R. Newcombe. The Replica Dataset: A Digital Replica of Indoor Spaces. *arXiv preprint*, arXiv:1906.05797, 2019.

[170] R. Strudel, R. Garcia, I. Laptev, and C. Schmid. Segmenter: Transformer for Semantic Segmentation. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[171] J. Stückler and S. Behnke. Multi-Resolution Surfel Maps for Efficient Dense 3D Modeling and Tracking. *Journal of Visual Communication and Image Representation (JVCIR)*, 25(1):137–147, 2014.

[172] E. Sucar, S. Liu, J. Ortiz, and A. Davison. iMAP: Implicit Mapping and Positioning in Real-Time. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[173] C. Sun, M. Sun, and H. Chen. Direct Voxel Grid Optimization: Super-fast Convergence for Radiance Fields Reconstruction. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[174] L. Sun, Z. Yan, A. Zaganidis, C. Zhao, and T. Duckett. Recurrent-OctoMap: Learning State-Based Map Refinement for Long-Term Semantic Mapping with 3D-Lidar Data. *IEEE Robotics and Automation Letters (RA-L)*, 3(4):3749–3756, 2018.

[175] W. Sun, N. Sood, D. Dey, G. Ranade, S. Prakash, and A. Kapoor. No-Regret Replanning under Uncertainty. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.

[176] N. Sünderhauf, T. Pham, Y. Latif, M. Milford, and I. Reid. Meaningful Maps with Object-Oriented Semantic Mapping. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

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

[178] Y. Tan, A. Kunapareddy, and M. Kobilarov. Gaussian Process Adaptive Sampling Using the Cross-Entropy Method for Environmental Sensing and Monitoring. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2018.

[179] M. Tanner, S. Saftescu, A. Bewley, and P. Newman. Meshed Up: Learnt Error Correction in 3D Reconstructions. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2018.

[180] A. Tewari, J. Thies, B. Mildenhall, P. Srinivasan, E. Tretschk, Y. Wang, C. Lassner, V. Sitzmann, R. Martin-Brualla, S. Lombardi, T. Simon, C. Theobalt, M. Nießner, J. Barron, G. Wetzstein, M. Zollhöfer, and V. Golyanik. Advances in Neural Rendering. *Computer Graphics Forum*, 41:703–735, 2022.

[181] V. Tresp. A Bayesian Committee Machine. *Neural Computation*, 12:2719–41, 12 2000.

[182] A. Trevithick and B. Yang. GRF: Learning a General Radiance Field for 3D Representation and Rendering. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[183] S. Vasudevan, F. Ramos, E. Nettleton, and H. Durrant-Whyte. Gaussian Process Modeling of Large-Scale Terrain. *Journal of Field Robotics (JFR)*, 26(10):812–840, 2009.

[184] E. Vespa, N. Nikolov, M. Grimm, L. Nardi, P. Kelly, and S. Leutenegger. Efficient Octree-Based Volumetric SLAM Supporting Signed-Distance and Occupancy Mapping. *IEEE Robotics and Automation Letters (RA-L)*, 3(2):1144–1151, 2018.

[185] T. Vidal-Calleja, D. Su, F. De Bruijn, and J. Miro. Learning Spatial Correlations for Bayesian Fusion in Pipe Thickness Mapping. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2014.

[186] A. Viseras, D. Shutin, and L. Merino. Online Information Gathering Using Sampling-Based Planners and GPs: An Information Theoretic Approach. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[187] A. Viseras and R. Garcia. DeepIG: Multi-Robot Information Gathering with Deep Reinforcement Learning. *IEEE Robotics and Automation Letters (RA-L)*, 4(3):3059–3066, 2019.

[188] I. Vizzo, T. Guadagnino, J. Behley, and C. Stachniss. VDBFusion: Flexible and Efficient TSDF Integration of Range Sensor Data. *Sensors*, 22(3):1296, 2022.

[189] S. Vora, N. Radwan, K. Greff, H. Meyer, K. Genova, M. Sajjadi, E. Pot, A. Tagliasacchi, and D. Duckworth. NeSF: Neural Semantic Fields for

Generalizable Semantic Segmentation of 3D Scenes. *Trans. on Machine Learning Research (TMLR)*, 2022.

[190] J. Wang and J. Kim. Semantic Segmentation of Urban Scenes with a Location Prior Map Using Lidar Measurements. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2017.

[191] J. Wang, M. Chen, N. Karaev, A. Vedaldi, C. Rupprecht, and D. Novotny. VGGT: Visual Geometry Grounded Transformer. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[192] J. Wang and B. Englot. Fast, Accurate Gaussian Process Occupancy Maps via Test-Data Octrees and Nested Bayesian Fusion. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2016.

[193] K. Wang, F. Gao, and S. Shen. Real-Time Scalable Dense Surfel Mapping. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2019.

[194] Q. Wang, Z. Wang, K. Genova, P. Srinivasan, H. Zhou, J. Barron, R. Martin-Brualla, N. Snavely, and T. Funkhouser. IBRNet: Learning Multi-View Image-Based Rendering. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[195] Q. Wang, Y. Zhang, A. Holynski, A. Efros, and A. Kanazawa. Continuous 3D Perception Model with Persistent State. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[196] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud. Dust3r: Geometric 3D Vision Made Easy. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[197] X. Wang, S. Liu, X. Shen, C. Shen, and J. Jia. Associatively Segmenting Instances and Semantics in Point Clouds. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[198] T. Whelan, S. Leutenegger, R. Moreno, B. Glocker, and A. Davison. ElasticFusion: Dense SLAM without a Pose Graph. In *Proc. of Robotics: Science and Systems (RSS)*, 2015.

[199] L. Wu, C. Le Gentil, and T. Vidal-Calleja. VDB-GPDF: Online Gaussian Process Distance Field with VDB Structure. *IEEE Robotics and Automation Letters (RA-L)*, 10(1):374–381, 2025.

[200] K. Wurm, C. Stachniss, and W. Burgard. Coordinated Multi-Robot Exploration Using a Segmentation of the Environment. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2008.

[201] Z. Xu, R. Jin, K. Wu, Y. Zhao, Z. Zhang, J. Zhao, F. Gao, Z. Gan, and W. Ding. HGS-Planner: Hierarchical Planning Framework for Active Scene Reconstruction Using 3D Gaussian Splatting. *arXiv preprint*, arXiv:2409.17624, 2024.

[202] B. Yamauchi. A Frontier-Based Approach for Autonomous Exploration. In *Proc. of the IEEE Intl. Symp. on Computer Intelligence in Robotics and Automation (CIRA)*, 1997.

[203] D. Yan, J. Liu, F. Quan, H. Chen, and M. Fu. Active Implicit Object Reconstruction Using Uncertainty-Guided Next-Best-View Optimization. *IEEE Robotics and Automation Letters (RA-L)*, 8(10):6395–6402, 2023.

[204] Z. Yan, H. Yang, and H. Zha. Active Neural Mapping. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2023.

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

[206] A. Yilmaz and H. Temeltas. Self-Adaptive Monte Carlo Method for Indoor Localization of Smart AGVs Using LIDAR Data. *Journal on Robotics and Autonomous Systems (RAS)*, 122:103285, 2019.

[207] A. Yu, V. Ye, M. Tancik, and A. Kanazawa. pixelNeRF: Neural Radiance Fields from One or Few Images. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[208] T. Zaenker, C. Smitt, C. McCool, and M. Bennewitz. Viewpoint Planning for Fruit Size and Position Estimation. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2021.

[209] A. Zaganidis, L. Sun, T. Duckett, and G. Cielniak. Integrating Deep Semantic Segmentation into 3D Point Cloud Registration. *IEEE Robotics and Automation Letters (RA-L)*, 3(4):2942–2949, 2018.

[210] R. Zeng, W. Zhao, and Y. Liu. PC-NBV: A Point Cloud Based Deep Network for Efficient Next Best View Planning. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2020.

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

[212] Y. Zeng, Y. Hu, S. Liu, J. Ye, Y. Han, X. Li, and N. Sun. RT3D: Real-Time 3D Vehicle Detection in LiDAR Point Cloud for Autonomous Driving. *IEEE Robotics and Automation Letters (RA-L)*, 3(4):3434–3440, 2018.

[213] H. Zhan, J. Zheng, Y. Xu, I. Reid, and H. Rezatofighi. ActiveRMAP: Radiance Field for Active Mapping and Planning. *arXiv preprint*, arXiv:2211.12656, 2022.

[214] R. Zhang, H. Bong, and G. Beltrame. Active Semantic Mapping and Pose Graph Spectral Analysis for Robot Exploration. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2024.

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

[216] L. Zheng, C. Zhu, J. Zhang, H. Zhao, H. Huang, M. Nießner, and K. Xu. Active Scene Understanding via Online Semantic Reconstruction. *Computer Graphics Forum*, 38(7):103–114, 2019.

[217] S. Zhi, T. Laidlow, S. Leutenegger, and A. Davison. In-Place Scene Labelling and Understanding with Implicit Scene Representation. In *Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV)*, 2021.

[218] X. Zhong, Y. Pan, J. Behley, and C. Stachniss. SHINE-Mapping: Large-Scale 3D Mapping Using Sparse Hierarchical Implicit Neural Representations. In *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2023.

[219] B. Zhou, Y. Zhang, X. Chen, and S. Shen. Fuel: Fast UAV Exploration Using Incremental Frontier Structure and Hierarchical Planning. *IEEE Robotics and Automation Letters (RA-L)*, 6(2):779–786, 2021.

[220] Y. Zhou and O. Tuzel. VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[221] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. Oswald, and M. Pollefeys. NICE-SLAM: Neural Implicit Scalable Encoding for SLAM. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[222] R. Zurbrügg, H. Blum, C. Cadena, R. Siegwart, and L. Schmid. Embodied Active Domain Adaptation for Semantic Segmentation via Informative Path Planning. *IEEE Robotics and Automation Letters (RA-L)*, 7(4):8691–8698, 2022.

[223] M. Zwicker, H. Pfister, J. Van Baar, and M. Gross. EWA Volume Splatting. In *Proc. of Visualization*, 2001.

List of Figures

List of Tables