Abstract of thesis entitled

"High-Performance and Energy-Saving Autonomous Navigation System for Aerial-Ground Robots"

Submitted by

Junming WANG

for the degree of Master of Philosophy

at The University of Hong Kong

in May, 2024

Aerial-ground robots (AGRs) are increasingly recognized for their valuable roles in search, exploration, and rescue tasks. This is attributed to their exceptional mobility and long endurance, which enable them to seamlessly switch between aerial and ground modes, allowing for hybrid locomotion (i.e., flying and driving) in the above challenging tasks. The success of AGRs relies on an autonomous navigation system comprising a perception module, path planner, and controller. Specifically, the perception module uses sensors (e.g., Camera or LiDAR) to map the environment, the path planner searches collision-free paths from this map, and the controller executes these paths adaptively. Additionally, the ability of AGRs to switch modes highlights the need for energy-efficient design to improve their operational efficiency and lifespan in demanding tasks. Given their sensitivity to energy consumption, prioritizing ground paths can help conserve energy.

To achieve high-performance and energy-saving navigation, Existing sensor-based AGR navigation systems utilize depth cameras to sense their surroundings and create Euclidean Signed Distance Field (ESDF) maps. These maps support quick and effective path planning, allowing AGRs to find collision-free paths. While this approach is effective in open and unobstructed environments, it often struggles in cluttered or unknown areas, due to the narrow field of view in sensor-based mapping and the path planner's inherent flaws. In these scenarios, both performance and energy efficiency are compromised.

This thesis presents two pioneering navigation systems, AGRNav and HE-Nav, exemplifying high performance and energy efficiency while tackling unique research challenges. To address the limitations of the sensor-based method in unknown and occluded environments, we introduced AGRNav, a tailored autonomous navigation solution for AGRs. The key innovation is the integration of a lightweight, predictive-based network, SCONet, which enhances real-time obstacle prediction in occluded areas by utilizing contextual cues. This approach effectively mitigates the perceptual constraints typically faced by traditional sensor-dependent methods. Additionally, AGRNav incorporates a query-based method for map updates with a hierarchical path planner, facilitating rapid updates to grid maps and enabling more energy-efficient path planning.

Next, recognizing the redundancy in constructing ESDF maps in existing ESDF-based methods, we introduced HE-Nav, a novel system designed specifically for AGRs. HE-Nav incorporates LBSCNet for enhanced perception, which excels in predicting obstacle distribution in occluded areas, and AG-Planner for ESDF-free path planning. This combination eliminates the need for ESDF map construction, streamlining the navigation process. LBSCNet's superior obstacle prediction capabilities, coupled with AG-Planner's efficient trajectory generation, significantly advance the efficiency of AGR autonomous navigation in complex environments.

In conclusion, the two innovative systems we introduce tackle critical perception and planning challenges that have traditionally hindered AGRs from attaining high-performance and energy-efficient autonomous navigation in environments with complex occlusions. Furthermore, through extensive testing in simulated environments and on our custom-designed AGR, we have thoroughly validated our system's superior performance and energy efficiency, demonstrating its comprehensive advantages in navigating intricate scenarios.

(466 words)

High-Performance and Energy-Saving Autonomous Navigation System for Aerial-Ground Robots

by

Junming WANG

M.Phil. *HKU*

A Thesis Submitted in Partial Fulfilment of the Requirements for
the Degree of Master of Philosophy
at University of Hong Kong

May, 2024

For Mama and Papa

Declaration

I, Junming WANG, declare that this thesis titled, "High-Performance and Energy-Saving Autonomous Navigation System for Aerial-Ground Robots", which is submitted in fulfillment of the requirements for the Degree of Master of Philosophy, represents my own work except where due acknowledgement have been made. I further declare that it has not been previously included in a thesis, dissertation, or report submitted to this University or to any other institution for a degree, diploma or other qualifications.

Signed .....

Junming WANG

Acknowledgements

I extend my heartfelt gratitude to Prof.Heming Cui for his unwavering support and invaluable guidance throughout my master's journey. His profound expertise and insightful suggestions have been instrumental in shaping my research topic and writing process, for which I am immensely grateful.

I am particularly indebted to Tianxiang Shen, Zekai Sun, Xiuxian Guan, Dong Huang, Zongyuan Zhang, Tianyang Duan, Shixiong Zhao, and Shengliang Deng for their significant contributions to the collaborative work presented in this thesis. Their dedication and expertise have greatly enriched the depth and quality of my research. Moreover, I would like to acknowledge the support and camaraderie of my fellow researchers and friends. The stimulating discussions, shared laughter, and moments of solidarity have made this journey all the more rewarding and memorable.

From the bottom of my heart, I extend my sincere appreciation to all my beloved family members who have stood by me throughout this challenging yet meaningful journey. Their unwavering love, encouragement, and sacrifices have been the driving force behind my perseverance and success. I am forever grateful for their presence in my life.

Lastly, I would like to commend myself for the relentless dedication and positive attitude maintained throughout this arduous process. The resilience in the face of challenges, and the unwavering commitment to personal and academic growth have been the cornerstone of my achievements. I am proud of the person I have become and the milestones I have reached.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation and Background

In recent years, Unmanned Ground Vehicles (UGVs) [37, 45, 17, 23, 38] excel in reconnaissance and rescue tasks, navigating well on structured terrains but struggle in complex environments like forests or hills. Conversely, multicopters [74, 75, 76, 77, 41, 73] offer superior manoeuvrability in challenging terrains but face limitations in power efficiency, particularly when tasked with extensive exploration or long-haul deliveries. These operations leverage the mobility of multicopters but concurrently strain their operational endurance. The situation is exacerbated when these aerial vehicles are required to carry substantial payloads, a common expectation in mission-critical operations, further intensifying the power efficiency dilemma.

To address these problems, the integration of multicopters with ground vehicles presents a synergistic approach, capitalizing on the high power efficiency of ground vehicles while retaining the exceptional mobility of multicopters. This amalgamation led to the advent of Aerial-Ground Robots (AGRs) [31, 68, 71, 13, 70, 40, 69, 50, 54, 57, 24], which possess the unique capability of vertical takeoff and landing, setting a new benchmark in transportation and mission execution. AGRs operate in dual modes: driving and flight, enabling them to effortlessly surmount a myriad of obstacles, including ditches and water bodies. This dual-mode operation empowers AGRs to execute a wide array of tasks, from three-dimensional reconnaissance to defence, transportation, and rescue missions in multifaceted environments [35, 6]. This kind of robot will most likely become the ultimate goal of the development of the next generation of transportation tools.

To enable AGRs to move in the above scenarios, they need a high-performance and energy-saving navigation system covering perception, planning, and control to achieve autonomous driving within a 3D range. Many scholars have researched the autonomous navigation of AGRs. Existing AGRs navigation system [68, 13, 69] utilize sensors (e.g., cameras or LiDAR) to perceive surrounding environments and establish Euclidean Signed Distance Field (ESDF) maps, subsequently the path planner to search for collision-free trajectories that favour ground paths and only switch to the aerial mode when necessary (e.g., encountering impassable obstacles), thereby promoting energy efficiency.

Unfortunately, While these navigation systems have proven successful in structured indoor scenarios, they face two limitations when navigating in complex and occluded environments (e.g., disaster zones or wilderness areas).

Firstly, the perception module results in incomplete local maps (i.e., containing unknown areas) of the narrow field of view in sensor-based mapping. This not only generates paths with high collision risk but also prolongs moving time since redundant paths. Secondly, the existing AGR path planners are inefficient. Specifically, building the ESDF map generates redundant calculation times that do not meet the real-time requirements (i.e., planning time < 1 ms) of path planning since it takes up about 70% of the time [75], and obstacles only take up 30% of the entire space. Notably, the path planner's inefficiency stems from the above intrinsic shortcomings and the perception module's limitations in providing a local map. Fortunately, for the perception module, to solve the above problem and generate complete local maps for navigation, the emerging semantic scene completion (SSC) network [58, 5, 29] holds promise, as it accurately predicts obstacle distribution and semantics in occluded areas. For the planner module, an ESDF-free planning framework for muticopters is proposed, named ego-planner [75, 76, 77], which significantly reduces the planning time.

However, The above perception networks and UAV planners cannot be naively migrated to AGRs because of some potential challenges. To begin with, existing networks face a trade-off between completion accuracy and fast inference. Some use 3D convolution [78] to improve accuracy but are unsuitable for resource-limited AGR devices (e.g., Jetson Xavier NX or Raspberry Pi) to ensure fast inference. Others propose lightweight network structures [43] but with significantly reduced accuracy. At the same time, how to equip the network with the ability to capture contextual information and learn long-distance dependencies, especially the prediction of obstacle distribution in occluded areas, is also a challenge that needs to be solved when designing the network. During the planning phase, while *Zhou et al.* [75] devised an ESDF-free path planner for quadcopters, it failed to address AGR-specific requirements, particularly energy efficiency and dynamic constraints. Their flight-centric trajectory generation results in elevated energy consumption and the inherent non-holonomic constraints of AGRs make it impossible to naively migrate and use such planners [75, 76, 77].

1.2 Solutions

In response to the aforementioned challenges posed by existing perception networks and path planners, this thesis proposes a comprehensive AGR navigation system design. By combining the working characteristics of AGR, a dedicated perception network and path planner are designed to achieve a high overall navigation result. performance and energy saving.

This thesis covers two novel systems, namely AGRNav and HE-Nav. Specifically, In the first work, AGRNav (§3) is introduced as a bespoke autonomous navigation solution for AGRs, incorporating a lightweight semantic scene completion network

(SCONet) aimed at precise obstacle prediction by leveraging contextual cues and features within occluded spaces. This method effectively overcomes the perceptual limitations of traditional sensor-reliant strategies. It also integrates a query-based approach with a hierarchical path planner, enabling rapid grid map updates and energy-efficient route determination. AGRNav's main contribution lies in the perception network and map update strategy. This navigation system solves the current most advanced AGR navigation system: TABV's [68] dilemma of being unable to predict obstacles in occluded areas.

In the second work. HE-Nav (§4) emerges as the inaugural ESDF-free navigation system specifically designed for AGRs. It introduces the lightweight yet potent LB-SCNet for perception, capable of forecasting obstacle distributions in obscured areas. In the planning stage, the AG-Planner initially creates an obstacle-ignorant trajectory, subsequently utilizing the energy-conscious Kinodynamic A* algorithm for identifying collision-free guide trajectory segments amidst obstacles. This process involves estimating the gradient between colliding and non-colliding guide segments, effectively navigating around obstacles without resorting to ESDF calculations. The trajectory undergoes further refinement through a gradient-based spline optimizer and a post-refinement procedure, culminating in an energy-efficient, safe, and dynamically viable path. The trajectory, optimized for aerial-ground navigation, is then precisely executed by the controller. HE-Nav is a comprehensive improvement over perceptual networks and path planners, and to our best knowledge is the first high-performance, energy-efficient, and ESDF-free aerial-ground robot navigation system.

1.3 Thesis Overview

The remainder of the thesis is structured as follows: Chapter 2 discusses background and related work. Chapter 3 introduces AGRNav, a novel *efficient* and *energy-saving* AGRs navigation system. Chapter 4 details the design and implementation of HE-Nav, the first *high-performance*, *efficient* and *ESDF-free* navigation system tailored for AGRs. Finally, Chapter 5 concludes and envisions future work.

1.4 Related Publications

- **Chapter 3 :** Junming Wang, Zekai Sun, Xiuxian Guan, Tianxiang Shen, Zongyuan Zhang, Tianyang Duan, Dong Huang, Shixiong Zhao and Heming Cui*, "AGRNav: Efficient and Energy-Saving Autonomous Navigation for Air-Ground Robots in Occlusion-Prone Environments", IEEE International Conference on Robotics and Automation (ICRA), 2024.
- **Chapter 4 :** Junming Wang, Zekai Sun, Xiuxian Guan, Shen Tianxiang, Zongyuan Zhang, Tianyang Duan, Fangming Liu and Heming Cui*, "HE-Nav: A High-Performance and Efficient Navigation System for Aerial-Ground Robots in Occluded Environments", [Under Submitted].

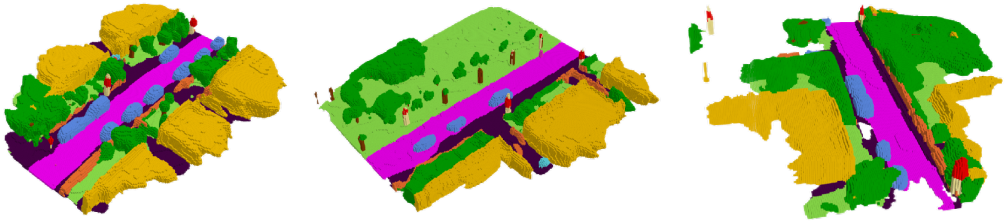
Chapter 2

Background

2.1 Occlusion-Aware for Aerial-Ground Robots (AGRs)

AGR's sensor-based perception method cannot make the local map include the distribution of obstacles in the occluded area, which will cause the planned path to be suboptimal. In recent years, the field of semantic scene completion [58, 78]has witnessed significant advancements, particularly in addressing the challenges posed by the obstruction of obstacles in complex and unknown environments. These advancements have led to the development of various point-cloud-based and camera-based methods. In the realm of camera-based methods, *Cao et al.* [5] introduced MonoScene, a groundbreaking approach that infers scene structure and semantics from a single monocular RGB image. Their key contribution lies in a novel 2D-3D feature projection bridging, inspired by optics, that enforces spatial semantic consistency. Another notable work by *Li et al.* [29] is VoxFormer, a Transformer-based semantic scene completion framework capable of generating complete 3D volume semantics using only 2D images. Yao et al. [66] introduced a novel approach, the Normalized Device Coordinates scene completion network (NDC-Scene), to improve Monocular 3D Semantic Scene Completion (SSC). They tackled existing issues like Feature Ambiguity and Computation Imbalance by extending 2D features to NDC space and using a Depth-Adaptive Dual Decoder for efficient 3D reconstruction from single images. Their method outperforms existing techniques on major datasets, demonstrating significant advancements in monocular SSC. *Li et al.* [30] present FB-OCC, a pioneering solution for 3D Occupancy Prediction, which stood out as the winning entry in a challenge associated with the CVPR 2023 workshops focused on autonomous driving. Their approach enhances the FB-BEV framework, which innovates in camera-based bird's-eye view perception through forward-backwards projection techniques. The team's advancements include integrating joint depth-semantic pre-training, a combined voxel-BEV representation, strategic model scaling, and refined post-processing methods. These improvements propelled their solution to achieve a leading mIoU score of 54.19% on the nuScenes dataset, securing the top position in the competition. *Yi Wei et al.* [56] proposed SurroundOcc, a novel approach for 3D occupancy prediction using multi-camera images, aimed at improving 3D scene understanding in autonomous driving. Their method extracts multi-scale

(a) NYUv2 Dataset



(b) SemanticKITTI Dataset

Figure 2.1: Popular datasets for Semantic Scene Completion (SSC).

features, employs spatial 2D-3D attention, and uses 3D convolutions for feature upsampling. They innovatively generate dense occupancy ground truth from LiDAR scans, avoiding costly annotations. Demonstrated on nuScenes and SemanticKITTI datasets, SurroundOcc achieves superior results, advancing vision-based autonomous driving research. *Huang et al.* [19] proposed a novel tri-perspective view (TPV) representation for vision-based 3D semantic occupancy prediction in autonomous driving, enhancing the traditional bird's-eye-view (BEV) by adding two additional perpendicular planes. Their approach, leveraging a transformer-based TPV encoder (TPVFormer), effectively aggregates image features into the 3D TPV space using an attention mechanism. Demonstrated on the nuScenes dataset, their method, trained with sparse supervision, matches the performance of LiDAR-based methods in LiDAR segmentation tasks, marking a significant advancement in camera-only perception systems.

On the other hand, point-cloud-based methods have also made significant strides. *Cheng et al.* [7] proposed S3CNet, a method designed to predict semantically complete scenes from a single unified LiDAR point cloud. *Roldao et al.* [43] introduced LMSCNet, a multiscale 3D semantic scene completion approach that uses a 2D UNet backbone with comprehensive multiscale skip connections to enhance feature flow, along with a 3D segmentation head. *Xia et al.* [58] devised a method that employs knowledge distillation from a multi-frame model to improve the performance of single-frame semantic scene completion. *Zuo et al.* [78] proposed PointOcc, a novel approach for point-based 3D semantic occupancy prediction in autonomous driving, utilizing a cylindrical tri-perspective view (TPV) to effectively represent point clouds. By constructing the TPV in a cylindrical coordinate system, their method allows for more detailed modeling of closer areas, crucial for LiDAR data. PointOcc employs spatial group pooling and 2D backbones for efficient processing, aggregating features from each TPV plane without

needing post-processing. Demonstrated on 3D occupancy prediction and LiDAR segmentation benchmarks, PointOcc significantly surpasses existing methods, including multi-modal approaches, on the OpenOccupancy benchmark, showcasing its state-of-the-art performance and speed. *Ming et al.* [39] proposed OccFusion, an efficient multi-sensor fusion framework for 3D occupancy prediction in autonomous driving. By integrating features from lidar, radar, and surround-view cameras, OccFusion enhances accuracy and robustness across diverse conditions, including challenging weather and lighting scenarios. Demonstrated on the nuScenes dataset, their approach achieves top-tier performance, especially in night and rainy conditions, showcasing the effectiveness of their sensor fusion strategy in improving 3D scene understanding. *Tarasha Khurana et al.* [26] proposed a novel approach for motion planning in autonomous systems by focusing on 3D point cloud forecasting from unannotated LiDAR sequences as a proxy for 4D occupancy forecasting. Their method addresses the limitations of classical methods that rely on costly human annotations by implicitly capturing sensor extrinsic, intrinsics, and the motion of objects. They recast the task to factor out sensor-specific details, enabling training and testing with unannotated data and facilitating evaluation across different datasets and sensors. This approach significantly advances the scalability and applicability of autonomous motion planning technologies.

Despite substantial progress in camera and point-cloud-based SSC methods, their high computational demands limit their suitability for resource-constrained AGR platforms. Thus, in the first system AGRNav, we proposed a lightweight semantic scene completion network (SCONet) to predict the distribution of obstacles in occluded regions. In the second system HE-Nav, we propose a lightweight SSC network using BEV feature fusion, serving as the perception module for the HE-Nav system, enabling rapid inference predictions and local map updates for path planning.

2.2 Motion Planning for Aerial-Ground Robots (AGRs)

Numerous researchers have explored various aerial-ground robot configurations, such as incorporating passive wheels [68, 57, 40, 42, 69, 52], cylindrical cages [24], or multi-limb [35] onto drones. In contrast, others [50, 71, 70, 54, 6] have integrated rotors with wheeled robots to achieve dual-mode (i.e., flying and driving) locomotion. These designs facilitate enhanced stability and control in both locomotion modes. Consequently, we also adopted this mechanical structure to customize further our AGR, which has four wheels and four rotors. Moreover, Existing research primarily focuses on innovative mechanical structure designs, and the area of AGR autonomous navigation remains underexplored. Recently, *Fan et al.* [13] address ground-aerial motion planning. Their approach initially employs the A* algorithm to search for a geometric path as guidance, favouring ground paths by adding extra energy costs to aerial paths. However, this path-searching method is limited due to its lack of dynamic models. Additionally, the local planner's trajectories lack post-refinement, resulting in potential issues with smoothness and dynamic feasibility. *Zhang et al.* [68] proposed a path planner and controller capable of efficient and adaptive path searching, but it relies on an

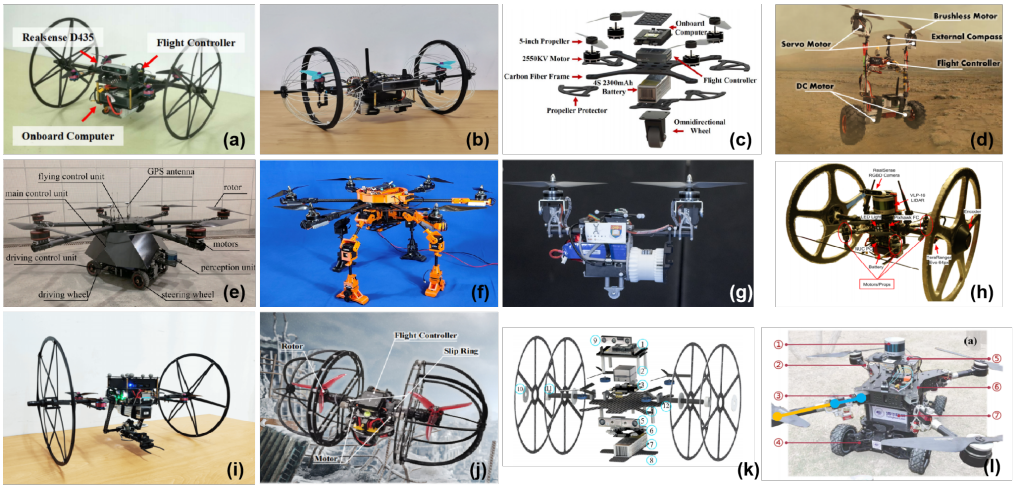


Figure 2.2: Different AGRs navigation systems have different mechanical structures.

ESDF map. The intensive computation and limited perception of occluded areas lead to a low success rate in path planning and increased energy consumption.

In this thesis, our AG-Planner eliminates the need for building ESDF maps by estimating the gradient of collision trajectory segments, significantly reducing computational effort. It employs an energy-efficient Kinodynamic A* search algorithm to find guidance paths and utilizes a gradient-based spline optimizer to obtain the optimal trajectory while considering collision, smoothing, and dynamic feasibility. A post-refinement process further improves trajectory robustness. Additionally, we address non-holonomic constraints by incorporating curvature limit costs for ground trajectories in the optimization formula.

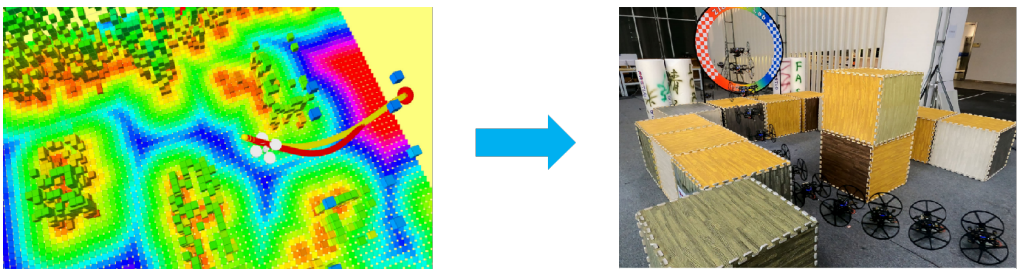


Figure 2.3: ESDF-based Air-Ground Robot Navigation System.

2.3 Energy-Efficient for Aerial-Ground Robots (AGRs)

Energy efficiency is vital for aerial-ground robots since it directly impacts their endurance and overall performance. By utilizing energy efficiently, these robots can operate for extended periods, reducing downtime and optimizing flight and ground navigation. Although the path planning frameworks proposed by *Fan et al.* [13] and *Zhang et al.* [68] take into account the energy consumption of AGRs, their approach is not comprehensive enough. They primarily add additional penalties to aerial flight, encouraging the robot to favour ground paths. While this strategy somewhat reduces

energy consumption, it overlooks other factors that contribute to energy inefficiency. For instance, when moving on the ground, the robot's turning angle can lead to additional energy consumption. Frequent steering demands extra energy from the motor, resulting in suboptimal energy usage. Phone Thiha Kyaw et al. [28] proposed an innovative algorithm, energy-efficient batch informed trees* (BIT*), for path planning in reconfigurable robots, focusing on energy efficiency and collision avoidance in complex environments. This approach enhances the BIT* planner with energy-based objectives for each reconfigurable action and introduces an L2 greedy informed set for improved sampling efficiency. Validated through experiments with a tetromino-hinged-based robot, their method demonstrates superior energy efficiency and faster convergence compared to existing techniques, in both simulated and real-world scenarios. Rafal Szczepanski et al. [49] proposed a novel energy-efficient local path planning algorithm that enhances the Artificial Potential Field (APF) method by incorporating future movement prediction and a unique local minimum avoidance technique using virtual obstacles called top quarks. This Predictive Artificial Potential Field (PAPF) algorithm significantly improves the smoothness of autonomous ground vehicle (AGV) movement, reduces travel distance, and efficiently bypasses local minima. Tested on the Husarion ROSbot 2.0 PRO, the PAPF algorithm demonstrated a reduction in electric power usage by 21.4%, shortened the path length by up to 8.73%, and decreased the time to reach the goal by up to 40.23%, outperforming the traditional APF method.

Therefore, we propose an energy-efficient Kinodynamic A* path search algorithm that comprehensively considers ground steering energy consumption and aerial flight energy consumption to search for dynamically feasible aerial-ground hybrid trajectories.

2.4 Map Updates for Aerial-Ground Robots (AGRs)

Recent studies have demonstrated the effectiveness of autonomous navigation systems that predict the distribution of obstacles in occluded areas to minimize collisions and conserve energy. However, these methods often fall short in complex environments and during high-speed operations. For example, the research by Katyal et al. (2021) [25] presents advanced perception algorithms and a control system that utilizes predicted occupancy maps for rapid navigation. Yet, this approach encounters difficulties in intricate, obstacle-rich settings due to its basic scene construction and infrequent map updates (around 3 Hz). In another study, Elhafsi et al. (2020) [12] introduce a network based on conditional neural processes for anticipating map routes, but their motion planning relies on heuristic methods that often result in suboptimal trajectories in uncharted territories. Furthermore, Wang et al. (2021) [53] developed OPNet, a technique for forecasting occupancy grids to facilitate path planning. While effective in straightforward scenarios, this method struggles in extensive occluded environments as its network cannot adequately understand the features and context of hidden areas.

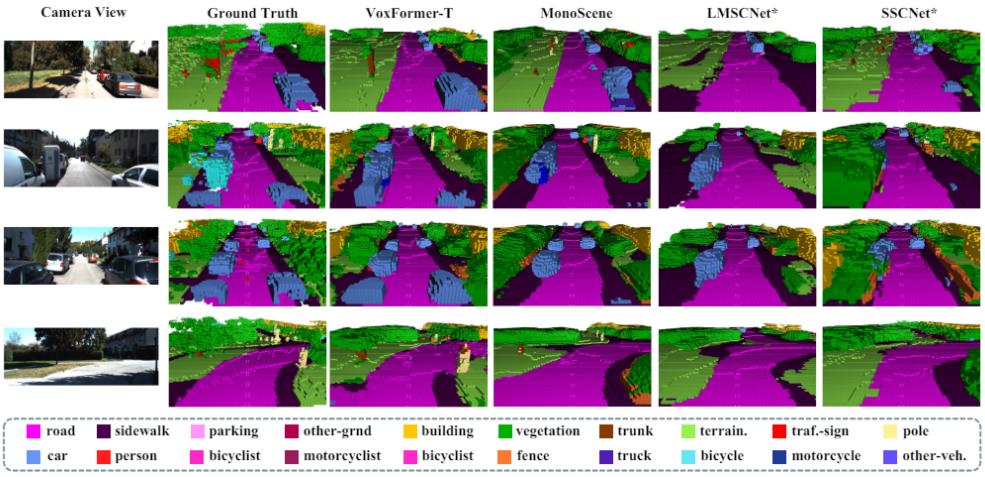


Figure 2.4: State-of-the-art camera-based 3D semantic scene completion network visualization results.

2.5 Related Work

The growing fascination with the flexibility and utility of AGR has sparked an increase in research and development within the sector. While many studies focus on optimizing the mechanical architecture of these robots to reduce their weight and size [68, 42, 13, 40], it is imperative to also develop an effective and energy-efficient navigation system that enables AGRs to operate in intricate settings. There is still considerable potential for enhancement in the navigation systems used by current air-ground robots.

In terms of perception, the limited field of view provided by sensors such as LiDAR and depth cameras poses challenges in covering occluded regions effectively. Research efforts predominantly concentrate on semantic scene completion techniques to predict the occupancy of these hidden areas. For instance, SSCNet [47] by *Song et al.*, utilizes depth imagery to forecast both the occupancy and semantics of voxels. Similarly, Monoscene [5] by *Cao et al.*, employs a single RGB image and introduces an innovative 2D-3D feature projection bridge for similar predictions. Despite their advancements, these approaches demand substantial memory, often exceeding 10 GB during inference with tools like Monoscene [5] and VoxFormer [29], rendering them impractical for real-time applications on robotic platforms due to their high resource consumption.

For planning, [13] presented air-ground path planning work, but due to the absence of trajectory refinement methods, the resulting trajectories lack smoothness and dynamic feasibility. [68] proposed an energy-saving and fast autonomous navigation framework, but its “aggressive" planning strategy increases the risk of collision when navigating complex and occluded areas. [57] propose an NMPC-based planning approach for Hybrid Terrestrial and Aerial Quadrotors (HyTAQs) aimed at enabling safe and efficient hybrid locomotion in unknown environments. By integrating complementarity constraints for hybrid dynamics, their approach facilitates simultaneous optimization of full-state trajectories and locomotion modes, avoiding the computational complexities associated with mixed-integer programming. The method incorporates

uncertainty bounds within geometry constraints in a receding horizon manner, enhancing safety and robustness. Additionally, it leverages topology-guided path searching to fully exploit bimodal energy efficiency, leading to more effective mode selection. The approach is demonstrated on a HyTAQ through extensive experimental evaluation, showcasing its high efficiency and robustness in navigating complex environments.

Chapter 3

AGRNav: Efficient and Energy-Saving Autonomous Navigation for Air-Ground Robots in Occlusion-Prone Environments

3.1 Introduction

Air-ground robots (AGR), which are known for their outstanding mobility and long endurance, have been gaining significant interest lately and show great potential for applications in search and rescue tasks [24, 40, 42]. Existing works [68, 13, 48] have demonstrated success in fast air-ground hybrid path planning, particularly in simple and unobstructed scenarios. However, AGR navigating complex environments (e.g., forests or buildings) with occluded and unknown areas faces a dilemma since obstacles in these areas significantly affect the results of path planning, i.e., high collision probability and suboptimal energy consumption (in Figure.3.1a).

To enable efficient and energy-saving navigation for air-ground robots in occluded environments, existing ***mapping-based*** methods [13, 68] use sensors (e.g., cameras or LiDAR) to construct a local occupancy grid map and an Euclidean Signed Distance Field (ESDF) map [74] for fast path planning. However, since the sensors' limitation is perceiving only visible obstacles (in Figure.1a), the constructed maps exclude obstructions in occluded areas, which increases the risk of collisions and leads to higher energy consumption from unnecessary aerial paths.

In contrast, existing ***learning-based*** methods employing semantic scene completion networks [5, 29, 47] to predict obstacle distribution in occluded areas and then enable the path planner to reduce unnecessary paths to achieve energy savings. Some networks use 3D convolutions [51] for enhanced prediction accuracy; however, their memory-intensive nature and high inference latency make them unsuitable for real-time robotic applications. While some work [53, 43] focuses on developing lightweight networks and achieving success in real-time inference, the network's limited ability to capture features and contextual information makes its prediction accuracy drop significantly. Additionally, addressing the update delay issue is also important, as delays may cause the path planner to ignore predicted obstacle distribution, thereby leading to similar problems as in mapping-based methods.

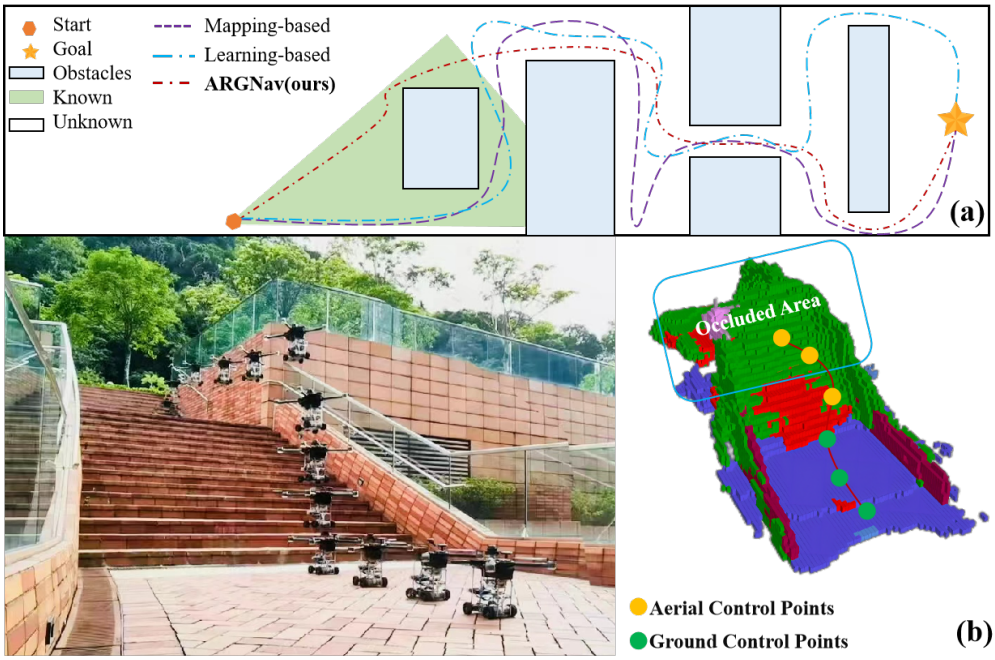


Figure 3.1: **(a)** Previous navigation systems had problems predicting occlusions, resulting in higher collision probabilities and suboptimal pathways that consumed more energy. **(b)** By predicting occlusions in advance, AGRNav can minimize and avoid collisions, resulting in efficient and energy-saving paths.

To tackle the high inference latency of memory-intensive networks and the low prediction accuracy of lightweight networks due to their inability to capture useful features, our key observation involves integrating lightweight convolutions and self-attention mechanisms into the network. The former allows the network to perform real-time inference tasks on robotic devices, while the latter enhances the network's ability to learn long-distance dependencies and capture contextual information, which is beneficial for improving the accuracy of prediction. Moreover, regarding update delay issues in existing methods that depend on map merging and result in repeated updates of occupied voxels, one potential method is only querying and updating the occupancy status of free voxels after scanning to ensure low latency.

Based on the above observations, we present ***AGRNav***, a novel efficient and energy-saving path-planning framework. The framework consists of two key components, the first one is a lightweight semantic scene completion network (SCONet), which is deployed on AGR and performs fast inference to accurately predict obstacle distribution and semantics. SCONet processes 3D voxel grids using depth-separable convolutions [8] rather than 3D convolutions, which greatly decreases the number of calculations. Furthermore, to enable SCONet to capture rich and dense contextual information as well as features of occlusion areas, it integrates two self-attention mechanisms. This keeps the network lightweight while enhancing its feature extraction capabilities (Figure.3.1b).

The hierarchical path (i.e., aerial and ground path) planner (in Figure. 3.2) utilizes a query-based method for low-latency occupancy updates. With the accurate predictions of SCONet, the planner minimizes collisions and energy consumption while searching for paths on an updated map containing scanned and predicted obstacles. Furthermore, it offers speed compensation for the robot using the semantics predicted, allowing for acceleration in passable areas, e.g., roads.

Simulations and real-world experiments show that the AGRNav enable search for safe and energy-saving pathways in occlusion-prone environments. The following are the key contributions of this paper:

- **AGRNav is efficient.** AGRNav achieves a 98% success rate in occlusion environments while also being low-latency in updating prediction results to the grid map.
- **AGRNav is energy-saving.** By predicting obstacle distribution in advance, unnecessary aerial paths are substantially reduced, resulting in a 50% decrease in energy consumption compared to the baseline.
- **SCONet is lightweight and accurate.** SCONet enables real-time (i.e., 20 FPS) and accurate inference and achieves state-of-the-art performance (IoU = 56.12) on the SemanticKITTI benchmark.

3.2 Related Work

3.2.1 Autonomous Navigation of Air-Ground Robots

The escalating interest in the adaptability and versatility of AGR has led to a surge of research and innovations in the field. Although many researchers prioritize mechanical structure design [68, 42, 13, 40] to minimize weight and volume, it is crucial to acknowledge that establishing an efficient and energy-saving navigation framework that empowers AGR to navigate in complex environments carries greater significance. Despite this, there remains room for improvement and further investigation in current air-ground robot navigation frameworks. For example, [13] presented air-ground path planning work, but due to the absence of trajectory refinement methods, the resulting trajectories lack smoothness and dynamic feasibility. [68] proposed an energy-saving and fast autonomous navigation framework, but its “aggressive" planning strategy increases the risk of collision when navigating complex and occluded areas.

3.2.2 Navigation in Predicted Maps

Autonomous navigation with low collision probability and energy savings by predicting obstacle distribution in occluded areas has shown promising results in recent studies. However, existing methods face limitations in complex environments and high-speed navigation scenarios. For instance, [25] introduces novel perception algorithms and a controller that incorporate predicted occupancy maps for high-speed navigation. Despite its potential, the method struggles to handle complex and obstacle-dense environments due to simplistic scene design and a lower map update frequency ($\approx$

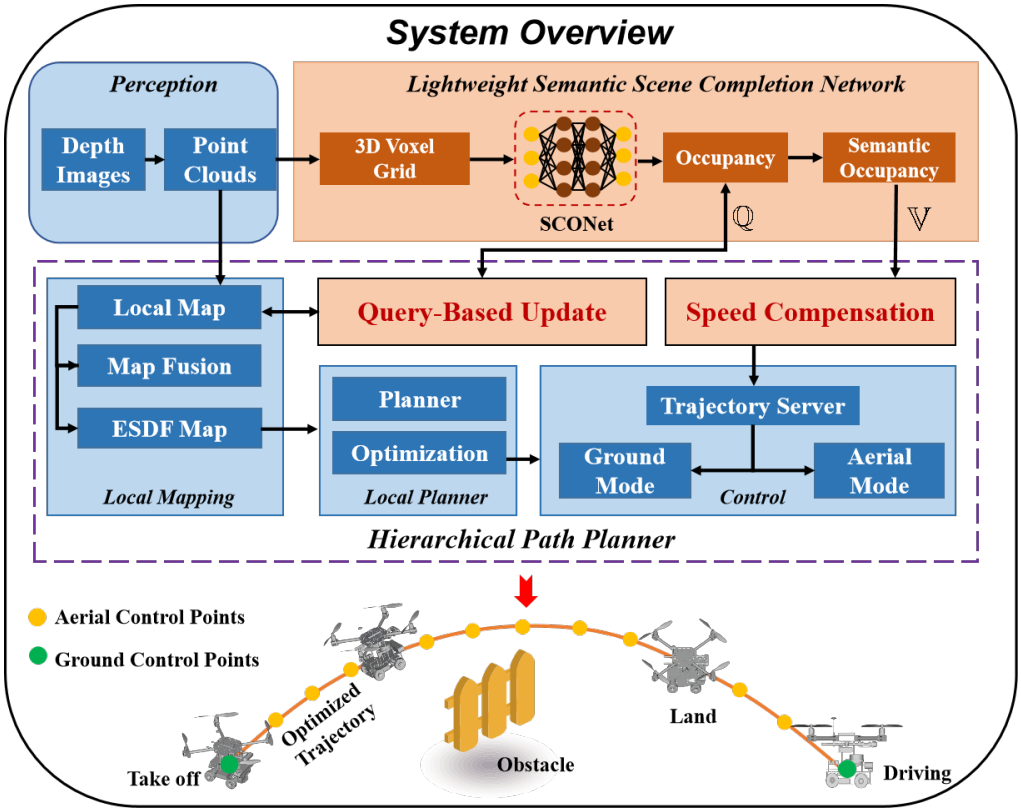


Figure 3.2: The overview of our proposed Framework: AGRNav. ℚ denotes that the free voxels in the grid map query and update their occupancy status from the predicted occupancy map. 𝕍 denotes that predicted semantics is turned into speed compensation.

3 Hz). Similarly, [12] employs a conditional neural process-based network to predict map turns but relies on heuristic approaches for motion planning in unknown environments. This results in greedy and inefficient trajectories without considering the unobserved environment's structure. Lastly, [53] proposed OPNet, a method that predicts occupancy grids for path planning and performs well in simple environments. However, the method faces challenges in large-scale occluded scenes because its network does not have the ability to capture the characteristics and contextual information of occluded areas. Gang Chen et al. [7] proposed a novel approach to address the limitations of traditional particle-based dynamic occupancy maps by introducing a continuous space model. They developed a 3-D egocentric local map using a dual-structure subspace division method, which combines voxel subspace division and a new pyramid-like subspace division, to efficiently propagate particles and update the map while accounting for occlusions. The method allows for the estimation of occupancy status at any point in the map space based on particle weights. To minimize noise in modeling both static and dynamic obstacles, the authors employed an initial velocity estimation technique and a mixture model. Experimental results demonstrated that their map significantly enhances the mapping of dynamic and static obstacles with improved performance across various resolutions, offering a substantial advancement

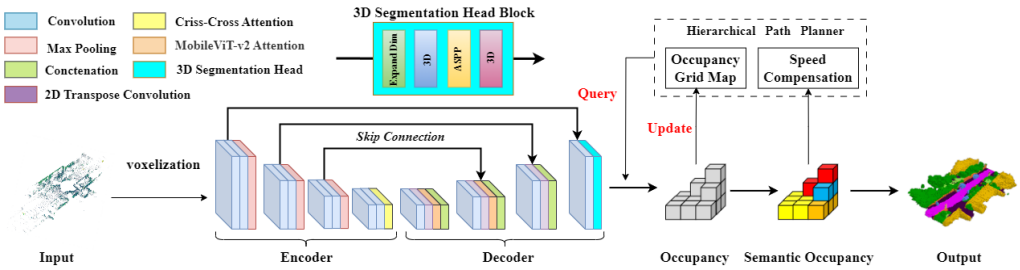


Figure 3.3: SCONet: Lightweight Semantic Scene Completion Network.Our network employs a self-attention-driven U-Net architecture, featuring depthwise separable convolutions and segmentation heads, to perform efficient 3D scene completion and semantic segmentation.

over traditional grid-form particle-based maps.

3.2.3 Semantic Scene Completion and Occupancy Mapping

Robot sensors with narrow fields of view, such as LiDAR and depth cameras, make it difficult to monitor occluded areas. The majority of the research on predicting occluded region occupancy using limited sensor data has focused on semantic scene completion approaches. Notable works include SSCNet [47] by *Song et al.*, which uses depth images to predict occupancy and semantics for voxels. Monoscene [5] by *Cao et al.*, only requires monocular RGB images and leverages a novel 2D-3D feature projection bridge to predict occupancy and semantics for voxels. However, these memory-intensive methods are unsuitable for real-time inference on robots' devices since Monoscene [5] and VoxFormer's [29] GPU memory exceeds 10 GB during inference.

3.3 System Overview

Fig.3.2 illustrates the proposed framework ***AGRNav***, featuring some key components: (1) the lightweight semantic scene completion network SCONet (Section IV); (2) the query-based low-latency occupancy update method (Section V-A); (3) the hierarchical path planner (Section V-B) search air-ground hybrid paths on the updated map which contains scanned obstacles and predicted obstacles.

3.4 Semantic scene completion network

3.4.1 SCONet Network Structure

We proposed a lightweight semantic scene completion network (SCONet) to predict the distribution of obstacles in occluded regions, as shown in Fig. 3.3. Point clouds are transformed into a 3D sparse voxel grid, serving as input for our 4-level U-Net style network. Each voxel is assigned a semantic label $\mathcal{L} = [i, l_1, l_2, ..., l_N], i = 0, 1$, where N is the number of semantic classes, $i = 0$ represents free voxels, and $i = 1$ represents occupancy voxels. This design allows for the effective prediction of obstacle distribution and their corresponding semantics using partial scans. The specific encoder and decoder structures are as follows:

Encoder. Instead of using memory-intensive 3D convolutions [51], we use considerably lighter depthwise separable convolutions [8] along the X and Y dimensions in the encoder, changing the height dimension (Z) into a feature dimension. This design learns features at a lower resolution while allowing direct processing of 3D voxel grids.

Decoder. By employing deconvolutions in the decoder, we up-sample the feature maps and subsequently concatenate output results to lower levels, which enhances the information flow while enabling our network to learn high-level features from coarser resolutions. Lastly, the semantics will be predicted through a 3D segmentation block that has a series of dense and dilated convolutions [67].

3.4.2 Two GPU Memory-Efficient Self-attention Mechanisms

The above design makes SCONet suitable for deployment on robotic devices for real-time inference. However, the network lacks the ability to capture contextual information and features in occluded areas, which is essential for improving prediction accuracy in such areas. Therefore, we have integrated two self-attention mechanisms into our architecture: *Criss-Cross Attention (CCA)* [20] and *MobileViT-v2 Attention* [36]. CCA, which is positioned after the encoder (in Fig. 3.3), enhances the network's ability to learn long-distance dependencies by collecting contextual information in horizontal and vertical directions. This enables the establishment of connections between distant features, which leads to more effective predictions of obstacles and semantics in occluded environments by comprehending the relationships among various elements (e.g., roads and walls).

Specifically, CCA takes the feature map $\mathbf{H} \in \mathbb{R}^{C \times W \times H}$ output from the fourth convolutional layer of the encoder as input, and then through two convolutional layers with 1×1 filters to acquire feature maps Q, K and attention maps A via affinity operation. Affinity can be defined as follows:

$$d_{i,u} = \mathbf{Q_u} \mathbf{\Omega_i}, \mathbf{u^T} \tag{3.1}$$

where $\mathbf{Q}, \mathbf{K} \in \mathbb{R}^{C^{'} \times W \times H}$, $\mathbf{A} \in \mathbb{R}^{(H+W-1) \times (W \times H)}$, Q_u is a vector at each position u in the spatial dimension of feature maps Q, Ω_u is a set combined all feature vectors from K which are in the same row or column with position u. The contextual information is collected by an aggregation operation defined as follows:

$$\mathbf{H}_u^{'} = \sum_{i=0}^{H+W-1} \mathbf{A}_{i,u} \mathbf{\Phi}_{i,u} + \mathbf{H}_u \tag{3.2}$$

where $\mathbf{H}_u^{'}$ is a feature vector at position u and i means channel. The contextual information is added to local feature H to augment the voxel-wise representation. The CCA's ability to learn long-distance dependencies enables SCONet to effectively understand the context and relationships between various structural elements in the environment, resulting in accurate obstacle prediction. To further achieve finer-grained semantic scene completion, such as trees and cars, and better capture features of regions in occluded areas, which in turn contributes to the reduction of robot collision probability, we integrated MobileViT-v2 Attention [36] into the first (resolution 1:8) and second

(resolution 1:4) layers of SCONet's decoder. This integration allows the extraction of diverse resolution fine-grained features, which further enhances the completion of areas of occluded regions, i.e., improves obstacle prediction accuracy. With a latency of just 3.4 ms [36], MobileViT-v2 Attention allows SCONet to maintain stronger feature capture capabilities while remaining lightweight. Mathematically, MobileViT-v2 Attention [36] can be defined as:

$$\mathbf{y} = \left(\underbrace{\sum \left(\overbrace{\sigma(\mathbf{x}W_I)}^{c_\mathbf{s} \in \mathbb{R}^k} * \mathbf{x}W_K \right)}_{c_\mathbf{v} \in \mathbb{R}^d} * \mathrm{ReLU}(\mathbf{x}W_V) \right) W_O \tag{3.3}$$

where $\mathbf{x}$ as input and $*$ means broadcastable element-wise multiplication and $\sum$ means summation operations. $W_O \in \mathbb{R}^{d \times d}$ means linear layer with weights. A ReLU activation to produce an output $\mathbf{x}_V \in \mathbb{R}^{k \times d}$.

3.5 Safe Air-Ground Hybrid Path Planner

The hierarchical path planner, building on the aerial-ground integration proposed by Zhang et al. [68], adeptly merges a query-based occupancy update mechanism, kinodynamic trajectory searching methodologies, and a gradient-based spline optimizer. Our hierarchical planner facilitates the creation of energy-efficient hybrid trajectories and enhances overall planning efficiency.

3.5.1 Query-Based Low-Latency Occupancy Update

The SCONet network generates a predicted occupancy grid map with occupied and free voxels. Typically, this map is merged with scan-based occupancy grid maps to construct the ESDF map for planning. The time complexity of this merge operation is $O(N)$, where N is the number of voxels since it needs to traverse and combine information from both grid maps. To achieve efficient navigation and obstacle avoidance, we proposed a query-based update method with low latency. Specifically, $f(x, S_{\text{pred}})$ represents the query operation, which checks whether the voxel x exists within the predicted occupied voxel set S_{pred}. If x is predicted to be occupied (i.e., $x \in S_{\text{pred}}$), then $f(x, S_{\text{pred}}) =$ occupied; otherwise, the status of x remains free. By focusing on M relevant free voxels, where $M \leq N$, this method reduce the time complexity to $O(M)$.

$$S_{\text{updated}}(x) = \begin{cases} \text{occupied}, & \text{if } f(x, S_{\text{pred}}) = \text{occupied} \\ \text{free}, & \text{otherwise} \end{cases} \tag{3.4}$$

3.5.2 Efficient and Energy-saving Hierarchical Path Planner

Different from the rough path search method of *Fan et al.* [13], we also further optimize the trajectories (contains ground and aerial trajectories), that is, set the trajectories as a p_b degree uniform B-spine with control points $\mathbf{P} = \{\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2, ..., \mathbf{P}_N\}$. In particular, the optimization and generation of trajectories are mainly divided into ground and aerial trajectories. When optimizing ground trajectories, we assume that the AGR moves on

flat ground, so we only need to consider the two-dimensional motion control point, denoted as:

$$\mathbf{P}_g = \{\mathbf{P}_{t0}, \mathbf{P}_{t1}, \mathbf{P}_{t2}, \mathbf{P}_{t3}, ..., \mathbf{P}_{tM-1}, \mathbf{P}_{tM}\} \tag{3.5}$$

where $\mathbf{P}_{ti} = (x_{ti}, y_{ti}), i \in [0, M]$. Meanwhile, the aerial trajectory control points are denoted as: $\mathbf{P}_a$. We also use the following cost terms designed by *Zhou et al.* [74] to refine the trajectory:

$$f_1 = \lambda_s f_s + \lambda_c f_c + \lambda_f (f_v + f_a) \tag{3.6}$$

where $\lambda_s, \lambda_c, \lambda_f$ are weights for each cost terms. f_s, f_c, f_v and f_a are smoothness, collision cost, soft limits on velocity and acceleration. We set the AGR to move in the ground mode, its speed is parallel to the yaw angle. In addition, considering that our AGR adopts the Akaman structure if the trajectory is too curved, there will be a huge error, so we enforce a cost on $\mathbf{P}_g$ to limit the curvature of the terrestrial trajectory, the curvature at $\mathbf{P}_{ti}$ is defined as:

$$\mathcal{C}_i = \frac{\triangle \beta_i}{\mathcal{P}_{ti}} \tag{3.7}$$

where $\triangle \beta_i = |tan^{-1} \frac{\triangle y_{ti+1}}{\triangle x_{ti+1}} - tan^{-1} \frac{\triangle y_{ti}}{\triangle x_{ti}}|$. Therefore, this cost can be formulated as:

$$f_n = \sum_{i=1}^{M-1} F_n(\mathbf{P}_{ti}) \tag{3.8}$$

Lastely, the overall objective function is formulated as follows:

$$f_{total} = \lambda_s f_s + \lambda_c f_c + \lambda_f (f_v + f_a) + \lambda_n f_n \tag{3.9}$$

and we use a non-linear optimization solver $NLopt$[2] to solve this optimization problem. After path planning is completed, a setpoint on the generated trajectory is selected according to the current timestamp and then sent to the controller. The settings and selections of the aerial and ground setpoint are the same as in [68].

3.6 Experiments

We evaluate AGRNav's improvement by comparing it against two mapping-based approaches and one learning-based method in two simulated environments. Moreover, we test AGRNav in three complex real-world scenarios employing a custom robot, showcasing its energy-saving advantages in practical navigation. By documenting the average energy consumption per second for AGR amidst driving and flying, we also establish a foundation for energy usage evaluation in simulated tests. Ultimately, we analyze SCONet's accuracy and real-time performance on the SemanticKITTI dataset.

3.6.1 Simulated Air-Ground Robot Navigation

The simulated experimental setup includes a $20m \times 20m \times 5m$ square room and a $3m \times 30m \times 5m$ corridor, which are filled with random obstacles, leading to numerous occlusion spaces and unknown regions throughout the scene. The air-ground robot must navigate from the starting point to the destination, and the maximum speed does not exceed 2.5 m/s.

Quantitative Results. We conducted a comparative analysis of our AGRNav navigation framework against two mapping-based and one learning-based navigation method

Table 3.1: The average power consumption per second and overall operational duration of an AGR in different modes.

Mode	Average Power	Time (mins)
Fly	987.61 W	14
Hover	532.07 W	26
Ground	197.52 W	55

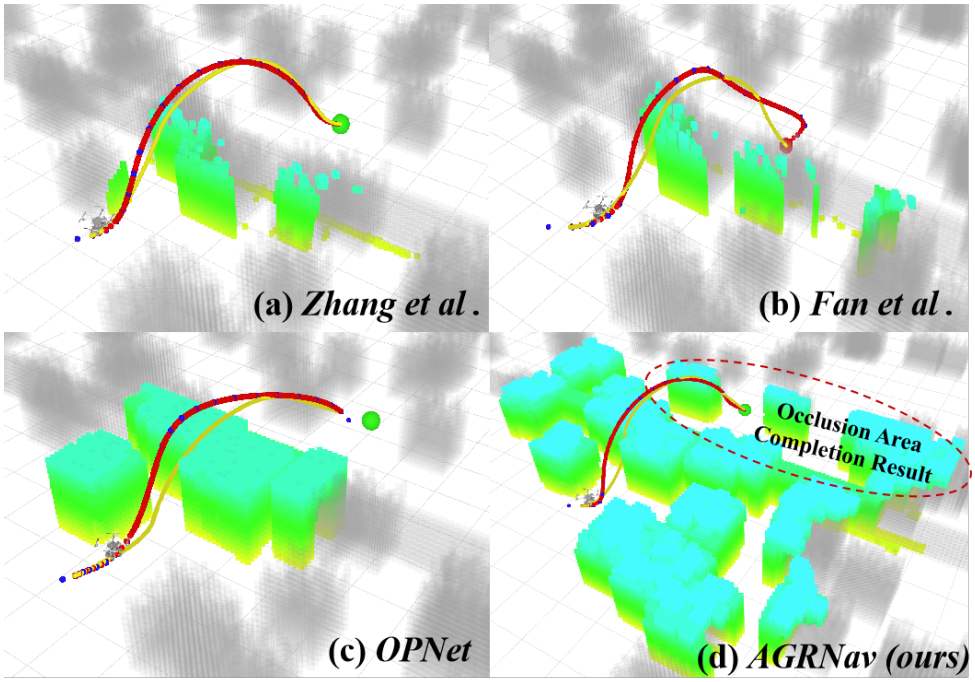


Figure 3.4: Four methods were used to plan paths in a simulated square room. AGRNav demonstrates the ability to predict the distribution of obstacles in occluded areas.

Table 3.2: Quantitative results in two simulation scenarios.

Env.	Method	Succ. (%)	Time (s)	Leng. (m)	Power (W)
Square Room	Fan's [13]	85.0	13.13	33.79	919.07
	Zhang's [68]	95.0	12.05	23.09	793.30
	OPNet [53]	91.0	12.90	32.12	888.04
	AGRNav(Ours)	**98.0**	**11.02**	**21.82**	**434.55**
Corridor	Fan's [13]	88.0	21.24	33.10	565.24
	Zhang's [68]	97.0	**16.97**	30.69	519.20
	OPNet [53]	90.0	18.45	32.85	534.11
	AGRNav(Ours)	**98.0**	17.50	**29.82**	**445.61**

in a square room and corridor scenario. 100 trials with varying obstacle placements, we recorded the average travel time, length and success rate (i.e., no collisions) for all 4 methods. In particular, the energy consumption of the four methods is calculated using the energy consumed per second by our customized robot when flying and driving in the real environment (Table 3.1). Table 3.2 shows that our AGRNav outperforms the other three approaches, achieving the highest success rate (i.e., 98%), since our network (SCONet) predicts a broader range of occlusion areas (in Figure.3.4d), and generates the path with the lowest collision rate.

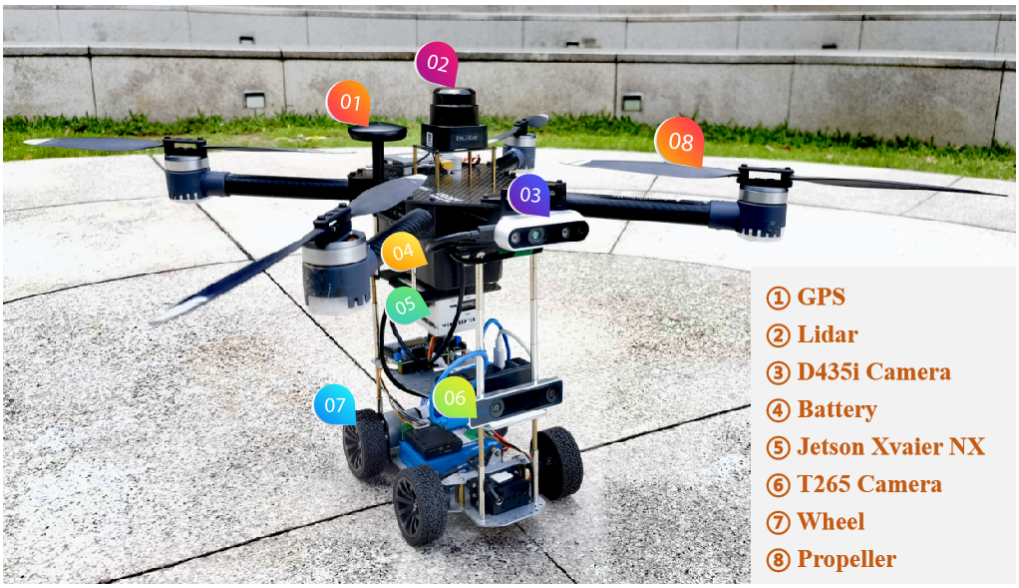


Figure 3.5: The detailed composition of our customized air-ground robot (AGR).

Furthermore, our framework substantially reduces redundant paths and cuts energy consumption by half (i.e., average consumption per second is 434.55 W) in a square room. This efficiency stems from SCONet's accurate predictions, which minimize high-energy-consuming aerial paths in favour of low-energy ground paths. In the corridor scene, while the average travel time of [68] is shorter (i.e., 16.97 s), its average energy consumption is higher due to the inability to predict occlusion areas and a greater reliance on aerial paths.

3.6.2 Real-world Air-Ground Robot Navigation

Our custom AGR platform (Figure. 3.5), is composed of a quadrotor with a 600mm diagonal wheelbase. This platform employs the Prometheus [1] software system and is equipped with a RealSense D435i depth camera and a T265 camera. It also features a Jetson Xavier NX onboard computer for the deployed AGRNav framework. Mobility is sustained by a 10,000 mAh energy source, which enables up to 26 minutes of hovering. Table 3.1 shows the energy consumption per second in different modes.

We evaluated the AGRNav's performance in 3 complex real-world environments where the robot's vision was obstructed by walls and bushes. In contrast to mapping-based methods that could result in potential collisions or suboptimal trajectories, our

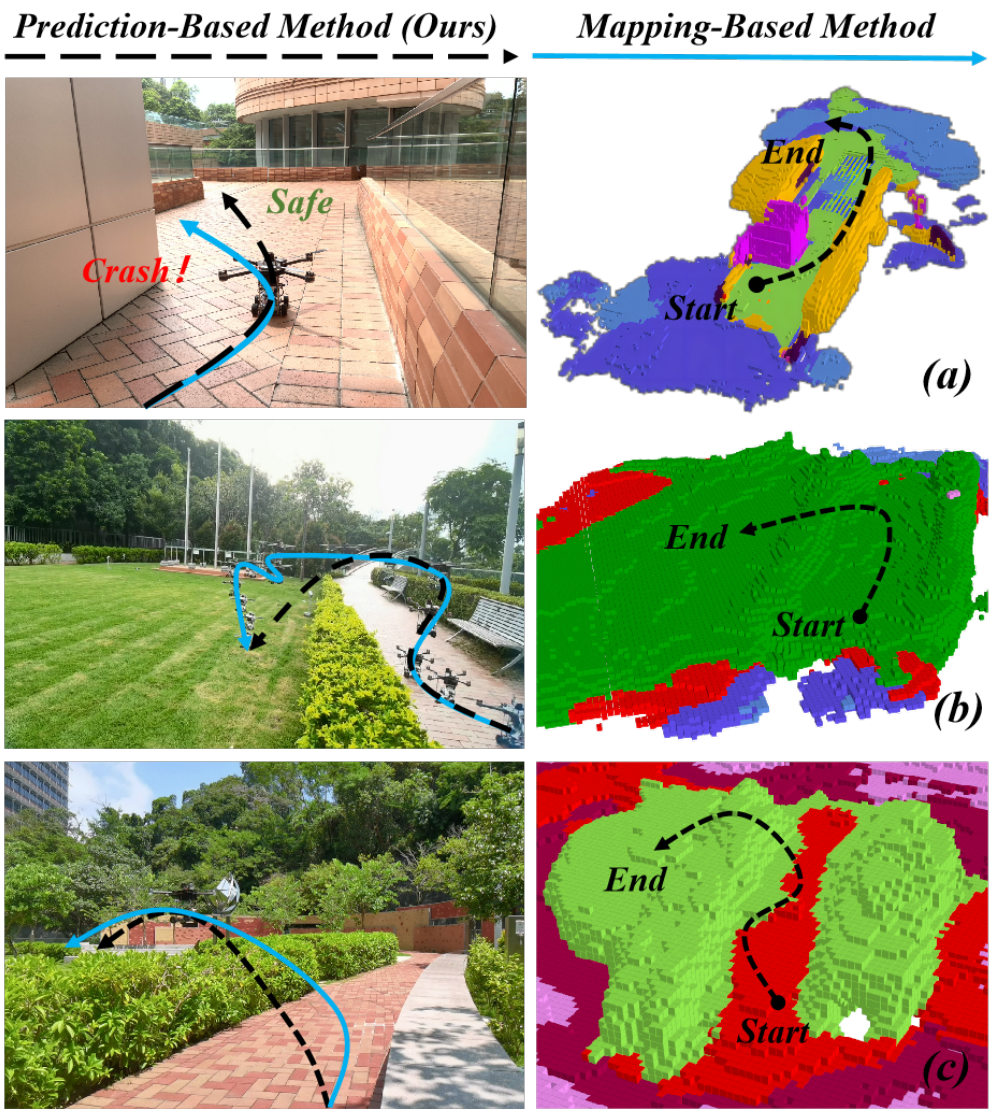


Figure 3.6: Navigation experiments of AGR in 3 complex real environments.

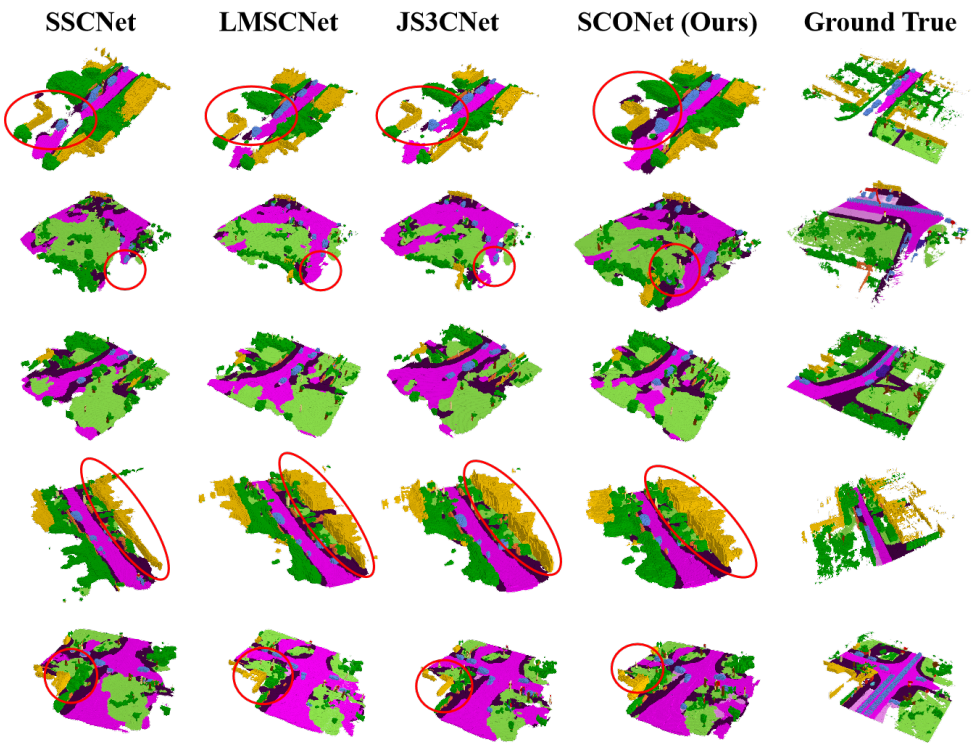


Figure 3.7: Qualitative results of SCONet on the validation set of SemanticKITTI.

AGRNav reliably predicted hidden obstacles (Figure. 3.6a), enabling safer navigation and searching energy-efficient paths (Figure. 3.6b) through SCONet's completion capabilities. Additionally, it identified optimal landing areas by foreseeing unseen obstacles (Figure. 3.6c), with semantic information aiding in velocity optimization for shorter travel times.

Table 3.3: Comparison of published methods on the official SemanticKITTI benchmark.

Method	*IoU*	*Prec.*	*Recall*	*FPS*	*mIoU*
SSCNet [47]	53.20	59.13	**84.15**	12.00	14.55
SG-NN [10]	31.26	31.60	54.50	12.00	9.90
J3S3Net [64]	51.10	40.23	61.09	1.73	23.80
LMSCNet [43]	54.89	82.21	62.29	18.50	14.13
S3CNet [7]	45.60	48.79	77.13	1.82	**29.50**
TDS [14]	50.60	72.43	78.61	1.70	17.70
SCONet (our)	**56.12**	**85.02**	63.47	**20.00**	17.61

3.6.3 Semantic Scene Completion Network (SCONet)

Pre-trained Model. We evaluate SCONet's performance using the SemanticKITTI benchmark [2], which offers 3D voxel grids from HDL-64 LiDAR scans in urban settings, labelled semantically. The input and ground truth grids are sparse, with dimensions of

Table 3.4: Ablation study of our model design choices on the SemanticKITTI [2] validation set.

Method	IoU ↑	mIoU ↑
SCONet (ours)	55.50	16.10
w/o Depth-Separable Convolution	54.15	15.76
w/o Criss-Cross Attention	53.20	15.11
w/o MobileViT-v2 Attention	53.86	15.37

$256 \times 256 \times 32$ and a $0.2m$ voxel size. Our analysis concentrated on completion metrics (IoU, precision, recall) and the semantic metrics mIoU, utilizing the benchmark's original splits and enhancing generalization through $x - y$ flipping augmentation. Employing the Adam optimizer at a learning rate of 0.001, scaled by 0.98 per epoch and conducted on a machine with 4 NVIDIA RTX3090 GPUs, training achieved convergence within 24 hours.

Quantitative Results. Table 3 reveals that our SCONet outperforms its rivals, registering the highest IoU completion metric score of 56.12. This result stems from testing on a hidden dataset via the official server. Despite a slightly lower mIoU compared to S3CNet and J3S3Net, SCONet's inference speed is significantly enhanced, being about 20 times faster. This is primarily due to the adoption of depthwise separable convolutions instead of the resource-heavy 3D convolutions in its encoder, enabling real-time efficiency with 20 FPS on an RTX 3090 GPU.

Qualitative Results. Fig. 7 illustrates that SCONet outshines the baseline model, notably enhancing the completion of structural objects like vehicles and trees (row 5). It adeptly handles completion in areas obscured by trees or walls (rows 1 and 4), a key feature for enabling efficient and safe path planning in later stages.

Ablation experiment results. Ablation studies on the SemanticKITTI validation set (Table 4) highlight the significance of two key components in our network: self-attention mechanisms and depth-separable convolutions. The CCA mechanism substantially impacts completion and semantic prediction by effectively aggregating context across rows and columns. *Without CCA* causes a 4.14% and 6.15% drop for completion and semantic completion, respectively. Meanwhile, MobileViT-v2 Attention captures local scene features, such as occluded areas, with low computational overhead. *Without MobileViT-v2 Attention* leads to a 2.95% decline in IoU. Furthermore, depth-separable convolutions significantly reduce the number of parameters.

3.7 Conclusions

In this paper, we introduce AGRNav, an efficient and energy-saving autonomous navigation framework for air-ground robots, featuring the key component SCONet, which outperforms state-of-the-art models in prediction accuracy and inference time. Additionally, a hierarchical path planner, improved by a query-based low-latency update method, considers obstacles in occluded areas to generate paths. This approach not only minimizes collision risk but also reduces energy consumption by 50% compared to the baseline by cutting down high-energy aerial paths. The system's robustness has

been extensively validated through experiments in both simulated and real-world environments.

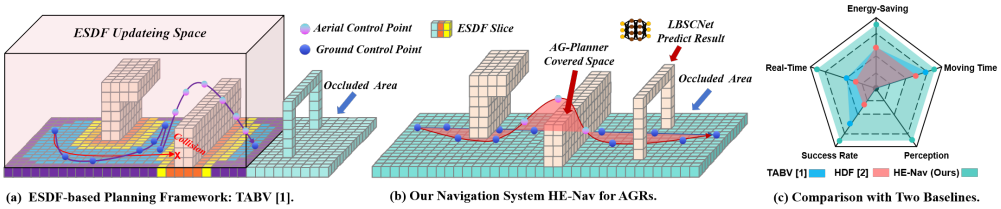
Chapter 4

HE-Nav: A High-Performance and Efficient Navigation System for Aerial-Ground Robots in Occluded Environments

4.1 Introduction

In recent years, aerial-ground robots (AGRs) [68, 13, 40, 69] have emerged as a promising solution for search [70, 71], exploration [46, 42], and rescue tasks [50, 54]. This is attributed to their exceptional mobility and long endurance, which enable them to seamlessly switch between aerial and ground modes, allowing for hybrid locomotion (i.e., flying and driving) in the above challenging tasks. Specifically, the ***perception module*** and the ***path planner*** are two crucial components in AGR navigation system that work synergistically, with the former generating a local map as the foundation for the latter to search for aerial-ground hybrid trajectories, ensuring *high-performance* (i.e., high planning success rate and shorter moving times) and *efficiency* (i.e., real-time planning and lower energy consumption).

Existing AGRs navigation system [68, 13, 69] utilize sensors (e.g., cameras) to perceive surrounding environments and establish Euclidean Signed Distance Field (ESDF) maps (in Fig. 4.1a), subsequently the path planner to search for collision-free trajectories that favour ground paths and only switch to the aerial mode when necessary (e.g., encountering impassable obstacles), thereby promoting energy efficiency.



(a) ESDF-based Planning Framework: TABV [1]. (b) Our Navigation System HE-Nav for AGRs. (c) Comparison with Two Baselines.

Figure 4.1: (a) ESDF-based methods face the dual predicaments of reduced path planning performance and increased energy consumption. **(b)** Our HE-Nav system can generate energy-saving, collision-free aerial-ground paths in real-time with the help of the LBSCNet model and AG-Planner. **(c)** HE-Nav Comparison with two baselines.

Unfortunately, While these ESDF-based navigation systems have proven successful in structured indoor scenarios, they face two limitations when navigating in complex and occluded environments (e.g., disaster zones or wilderness areas).

Firstly, the *perception module* results in incomplete local maps (i.e., containing occlusion induced unknown areas) since the narrow field of view in sensor-based mapping. This not only generates paths with high collision risk (e.g., *red path* in Fig.4.1a.) but also prolongs moving time since redundant paths (e.g., *purple path* in Fig.4.1a). To solve the above problem and generate complete local maps for navigation, the emerging semantic scene completion (SSC) network [58, 5, 29] holds promise, as it accurately predicts obstacle distribution and semantics in occluded areas. However, existing networks face a trade-off between completion accuracy and fast inference. Some use 3D convolution [78] to improve accuracy but unsuitable for resource-limited AGR devices (e.g., Jetson Xvaier NX) to ensure fast inference. Others propose lightweight network structures [43] but with significantly reduced accuracy.

Secondly, the existing AGRs *path planners* are inefficient. Specifically, building the ESDF map generates redundant calculation times that do not meet the real-time requirements (i.e., planning time $<$ 1 ms) of path planning since it takes up about 70% of the time [75], and obstacles only take up 30% of the entire space (in Fig. 4.1a). Moreover, while the energy cost of flying is considered, the energy implications of ground movement (e.g., steering) are often overlooked, leading to overall energy inefficiency. Notably, the path planner's inefficiency stems from the above intrinsic shortcomings and the perception module's limitations in providing a local map.

To tackle these above limitations, we present ***HE-Nav***, the first *high-performance*, *efficient* and *ESDF-free* navigation system tailored for AGRs, as illustrated in Fig. 4.2. Initially, we developed the lightweight LBSCNet as a perception module to predict obstacle distribution in occluded areas. By processing sparse point clouds, it produces voxel occupancy and semantics, which are subsequently integrated into the local map for path planning.

During the planning phase, our AG-Planner generates an ignore obstacles initial trajectory (i.e., *blue path* in Fig.4.1b.) and employs the energy-efficient Kinodynamic A* algorithm to search for corresponding collision-free guide trajectory segments within obstacles. We then estimate the gradient between colliding and collision-free guide segments, wrapping the trajectory around obstacles while avoiding unnecessary ESDF computations. Lastly, a gradient-based spline optimizer and post-refinement process further refine the aerial-ground trajectory, yielding an energy-efficient, safe, smooth, and dynamically feasible path (i.e., *brown path* in Fig.4.1b.). The optimized trajectory is subsequently sent to the controller for precise tracking.

However, the design of the LBSCNet and AG-Planner confronts multiple challenges. First, balancing completion accuracy and high-speed inference remains a challenge for LBSCNet. Despite our LBSCNet employing sparse 3D convolutions [16] for a lightweight structure, it inadequately captures contextual information in occluded areas, reducing accuracy, while 3D feature fusion raises inference latency, affecting path

planning performance.

To address these issues, we separate semantic and geometric learning processes into distinct branches, effectively leveraging their complementarity. Next, we incorporate the Criss-Cross Attention (CCA) [20] mechanism within the completion branch, enabling the sparse 3D convolutions can capture long-range dependencies and contextual information. Finally, we introduce the SCB-Fusion component, which facilitates the merging of BEV, semantic, and geometric features in the BEV space, ultimately reducing computational complexity and enhancing accuracy. (§ 4.3)

Secondly, while *Zhou et al.* [75] devised an ESDF-free path planner for quadcopters, it fails to address AGR-specific requirements, particularly energy efficiency and dynamic constraints. Their flight-centric trajectory generation results in elevated energy consumption and the inherent non-holonomic constraints of AGRs make it impossible to naively migrate and use such planners.

To overcome these challenges, our AG-Planner employs a novel energy-efficient Kinodynamic A* algorithm, which adds additional energy costs to motion primitives involving sharp ground turns or aerial destinations, thereby promoting energy efficiency. Concurrently, we account for AGRs' non-holonomic constraints by limiting ground control point curvature. We then utilize an obstacle distance estimation method from [75] to circumvent obstacles, avoiding ESDF computations. To the best of our knowledge, AG-Planner is the first ESDF-free and energy-efficient planner tailored for AGRs. (§ 4.4)

We first assessed LBSCNet on the SemanticKITTI benchmark, comparing its accuracy and inference speed to a leading SSC network. Then, we tested HE-Nav in simulated and real environments, contrasting it with two AGR navigation baselines, showcasing its superior performance and efficiency (Fig.4.1c). Our evaluation reveals:

- **HE-Nav is high-performance.** HE-Nav achieved success rates of 98% and 97% in the two simulation scenarios, respectively, while having the shortest average movement time. (§ 4.5.3)
- **HE-Nav is real-time planning.** AG-Planner achieves an **8x** reduction in planning time compared to ESDF-based methods. (§ 4.5.4)
- **HE-Nav is energy efficient.** AG-Planner significantly cuts energy consumption by 24.98% and 25.03% in two simulated settings, and by 10.34% in real-world outdoor situations. (§ 4.5.3 and § 4.5.4)
- **LBSCNet is accurate and high-speed inference.** LBSCNet achieves state-of-the-art performance (IoU = 59.71) on the SemanticKITTI benchmark and enables high-speed inference (20.08 FPS). (§ 4.5.2)

Our main contributions comprise the development of the lightweight LBSCNet and the energy-efficient ESDF-free AG-Planner. (1) LBSCNet, featuring innovative architecture and components such as the BEV fusion branch and SCB-Fusion module, enables rapid inference and complete local map generation. (2) Building upon this foundation, AG-Planner accomplishes ESDF-free planning with minimized planning

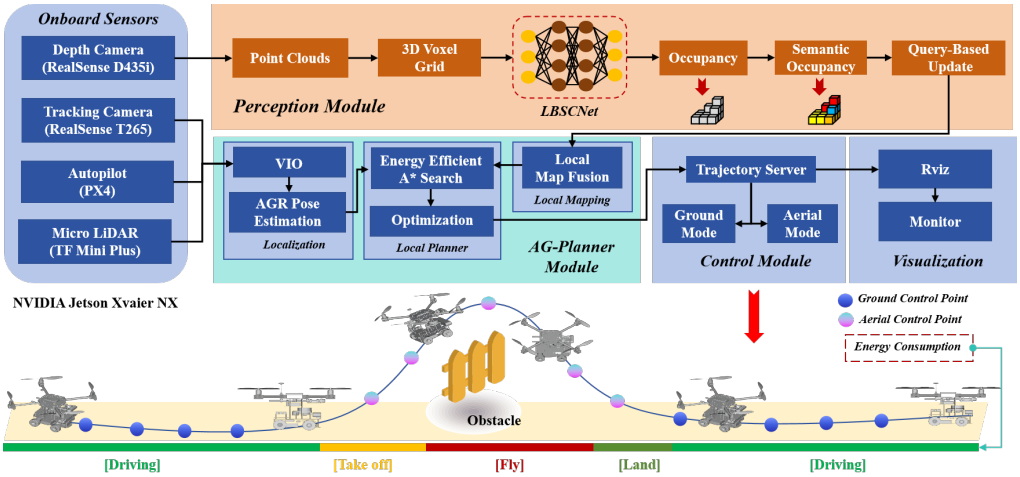


Figure 4.2: HE-Nav system architecture. The perception, planning and control modules are deployed on the onboard computer (i.e., NVIDIA Jetson Xavier NX) and run asynchronously.

time. Concurrently, by incorporating costs for ground control points and utilizing the energy-conscious Kinodynamic A* algorithm, our HE-Nav generates energy-efficient, safe, smooth, and dynamically feasible hybrid trajectories.

To the best of our knowledge, HE-Nav is the first AGR-tailored navigation system, combining occlusion awareness and ESDF-free aerial-ground hybrid path planning, ensuring high-performance and efficient autonomous navigation in occluded environments.

4.2 Related Work

4.2.1 Motion Planning for AGRs

Numerous researchers have explored various aerial-ground robot configurations, such as incorporating passive wheels [68, 57, 40, 42, 69, 52], cylindrical cages [24], or multi-limb [35] onto drones. In contrast, others [50, 71, 70, 54, 6] have integrated rotors with wheeled robots to achieve dual-mode (i.e., flying and driving) locomotion. These designs facilitate enhanced stability and control in both locomotion modes. Consequently, we also adopted this mechanical structure to customize further our AGR, which has four wheels and four rotors. Moreover, Existing research primarily focuses on innovative mechanical structure designs, and the area of AGR autonomous navigation remains underexplored. Recently, *Fan et al.* [13] address ground-aerial motion planning. Their approach initially employs the A* algorithm to search for a geometric path as guidance, favouring ground paths by adding extra energy costs to aerial paths. However, this path-searching method is limited due to its lack of dynamic models. Additionally, the local planner's trajectories lack post-refinement, resulting in potential issues with smoothness and dynamic feasibility. *Zhang et al.* [68] proposed a path planner and controller capable of efficient and adaptive path searching, but it relies on an ESDF map. The intensive computation and limited perception of occluded areas lead to a low success rate in path planning and increased energy consumption.

In this paper, our AG-Planner eliminates the need for building ESDF maps by estimating the gradient of collision trajectory segments, significantly reducing computational effort. It employs an energy-efficient Kinodynamic A* search algorithm to find guidance paths and utilizes a gradient-based spline optimizer to obtain the optimal trajectory while considering collision, smoothing, and dynamic feasibility. A post-refinement process further improves trajectory robustness. Additionally, we address non-holonomic constraints by incorporating curvature limit costs for ground trajectories in the optimization formula.

4.2.2 Occlusion-Aware for AGRs

AGR's sensor-based perception method cannot make the local map include the distribution of obstacles in the occluded area, which will cause the planned path to be suboptimal. In recent years, the field of semantic scene completion [58, 78]has witnessed significant advancements, particularly in addressing the challenges posed by the obstruction of obstacles in complex and unknown environments. These advancements have led to the development of various point-cloud-based and camera-based methods. In the realm of camera-based methods, *Cao et al.* [5] introduced MonoScene, a groundbreaking approach that infers scene structure and semantics from a single monocular RGB image. Their key contribution lies in a novel 2D-3D feature projection bridging, inspired by optics, that enforces spatial semantic consistency. Another notable work by *Li et al.* [29] is VoxFormer, a Transformer-based semantic scene completion framework capable of generating complete 3D volume semantics using only 2D images. On the other hand, point-cloud-based methods have also made significant strides. *Cheng et al.* [7] proposed S3CNet, a method designed to predict semantically complete scenes from a single unified LiDAR point cloud. *Roldao et al.* [43] introduced LMSCNet, a multiscale 3D semantic scene completion approach that uses a 2D UNet backbone with comprehensive multiscale skip connections to enhance feature flow, along with a 3D segmentation head. *Xia et al.* [58] devised a method that employs knowledge distillation from a multi-frame model to improve the performance of single-frame semantic scene completion.

Despite substantial progress in camera and point-cloud-based SSC methods, their high computational demands limit their suitability for resource-constrained AGR platforms. Thus, we propose a lightweight SSC network using BEV feature fusion, serving as the perception module for the HE-Nav system, enabling rapid inference predictions and local map updates for path planning.

4.2.3 Energy-Efficient for AGRs

Energy efficiency is vital for aerial-ground robots since it directly impacts their endurance and overall performance. By utilizing energy efficiently, these robots can operate for extended periods, reducing downtime and optimizing flight and ground navigation. Although the path planning frameworks proposed by *Fan et al.* [13] and *Zhang et al.* [68] take into account the energy consumption of AGRs, their approach is not comprehensive enough. They primarily add additional penalties to aerial flight,

encouraging the robot to favour ground paths. While this strategy somewhat reduces energy consumption, it overlooks other factors that contribute to energy inefficiency. For instance, when moving on the ground, the robot's turning angle can lead to additional energy consumption. Frequent steering demands extra energy from the motor, resulting in suboptimal energy usage.

Therefore, we propose an energy-efficient Kinodynamic A* path search algorithm that comprehensively considers ground steering energy consumption and aerial flight energy consumption to search for dynamically feasible aerial-ground hybrid trajectories.

4.3 Perception Module of HE-Nav

In this section, we introduce a lightweight three-branch SSC network (LBSCNet), depicted in Fig. 4.3. LBSCNet consists of a semantic branch, a completion branch, and a BEV fusion branch, serving as an alternative to conventional memory-intensive SSC networks that jointly predict geometry and semantics. By employing a pre-trained model offline on AGR devices, LBSCNet can infer and predict the obstacle distribution in occluded areas at high speed. Subsequently, these prediction results are updated into a local map, which is utilized for path planning.

4.3.1 LBSCNet Network Structure

LBSCNet decoupling the learning process of semantics and completion (or geometry), allows the network to concentrate on specific features (i.e., semantics and geometry), resulting in more efficient and fast learning. The specific structures are as follows:

***Semantic Branch*:** This branch consists of a voxelization layer and three encoder blocks sharing a similar architecture, each encoder block comprises a residual block [18] with sparse 3D convolutions and a cross-scale global attention (CSGA) module from [61]. The integration of the CSGA module not only aligns multi-scale features with global voxel-encoded attention to capturing the long-range relationship of context but also alleviates the computational burden by reducing feature resolution.

Specifically, in the voxelization layer, point clouds $P \in \mathbb{R}^{N\times 3}$ are partitioned based on the voxel resolution s and mapped into voxel space. Subsequently, an aggregation function (i.e., max function) is applied to the point cloud within each voxel, yielding a single feature vector. A multi-layer perceptron (MLP) reduces the dimensionality of this feature vector, producing the final voxel features V_{f_m} with a spatial resolution of $L \times W \times H$, f_m represents the index of the voxel. The voxel features V_{f_m} are then input into three encoder blocks to obtain semantic features $\{Sem_f^1, Sem_f^2, Sem_f^3\}$ (Fig. 4.3). The semantic branch is optimized using lovasz loss [3] and cross-entropy loss [72]. The semantic loss L_{sem} is the sum of the loss at each stage, expressed as follows:

$$L_{sem} = \sum_{i=1}^{3}(L_{cross,i} + L_{lovasz,i}) \tag{4.1}$$

***Completion Branch*:** The input to the completion branch is voxels $V \in \mathbb{R}^{1\times L\times W\times H}$

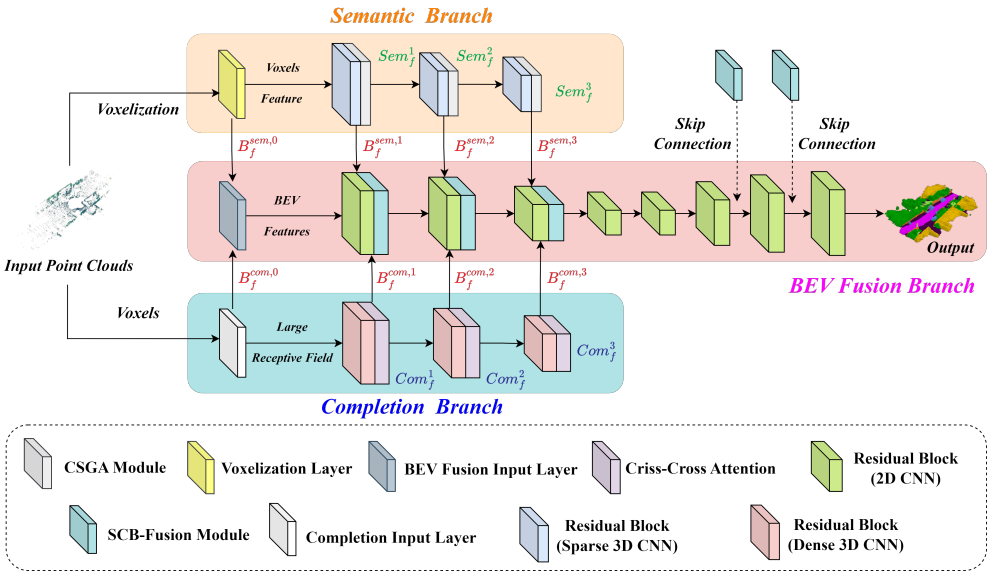


Figure 4.3: The overview of the proposed LBSCNet. It consists of semantic, completion and BEV fusion branches.

generated by point clouds. The output is the multi-scale dense completion features $\{Com_f^1, Com_f^2, Com_f^3\}$, providing more intricate geometric information.

As depicted in Fig. 4.3, the completion branch comprises an input layer (kernel size $7 \times 7 \times 7$), three residual blocks and three GPU memory-efficient criss-cross attention (CCA) [20] modules. The residual blocks incorporate dense 3D convolutions with a kernel size of $3 \times 3 \times 3$, capturing local geometric features. Conversely, the criss-cross attention (CCA) [20] module is designed to capture long-range dependencies by gathering contextual information in both horizontal and vertical directions, thereby enriching the completion features with a global context. The training loss L_{com} for this branch is calculated as follows:

$$L_{com} = \sum_{i=1}^{3} (L_{binary\_cross,i} + L_{lovasz,i}) \tag{4.2}$$

where i denotes the $i-th$ stage of the completion branch and L_{binary_cross} indicates the binary cross-entropy loss. Notably, during training, both the semantic and completion branches undergo deep supervision [33]. Lightweight MLPs are attached as auxiliary heads [61] after each encoder block to obtain semantic and geometric predictions for valid voxels. However, during inference, these auxiliary heads are removed to maintain a lightweight network structure.

***BEV Feature Fusion Branch*:** Previous research on SSC tasks has relied on fusing dense 3D features, resulting in considerable computational overhead and hindering deployment on resource-constrained AGR devices. We propose a lightweight BEV fusion branch specifically designed for SSC tasks, capitalizing on recent advancements in BEV perception [34, 65, 32]. By projecting learned semantic and geometric features into BEV space and incorporating the innovative SCB-Fusion module, we significantly reduce computational demands while maintaining rapid inference capabilities. Specifically,

our BEV fusion network employs a U-Net architecture with 2D convolutions, featuring an input layer and four residual blocks in the encoder (Fig. 4.3). The process of projecting semantic and geometric features to BEV space is as follows:

***Semantic Feature Projection*:** In order to project three-dimensional semantic features $\{Sem_f^1, Sem_f^2, Sem_f^3\}$ into the two-dimensional BEV space, we first generate a BEV index based on the voxel index f_m and then the features sharing identical BEV indices are aggregated using an aggregation function (e.g., the max function) to yield sparse BEV features. Utilizing the feature densification function offered by spconv [9], we generate dense BEV features $\{B_f^{sem,0}, B_f^{sem,1}, B_f^{sem,2}, B_f^{sem,3}\}$ based on the BEV index and sparse BEV features.

***Geometric Feature Projection*:** For geometric features $\{Com_f^1, Com_f^2, Com_f^3\}$, we stack dense 3D features along the z-axis and apply 2D convolution to reduce the feature dimension, generating dense BEV features $\{B_f^{com,0}, B_f^{com,1}, B_f^{com,2}, B_f^{com,3}\}$. Subsequently, the projected features are input into the BEV fusion network (Fig. 4.3). The BEV loss L_{bev} is :

$$L_{bev} = L_{cross} + L_{lovasz} \tag{4.3}$$

***Feature Fusion after Projection*:** To fuse the projected features, we devise an SCB-Fusion module (Fig. 4.13a) that fuses current semantic features, geometric features, and BEV features from the previous layer. Specifically, we first compute channel attention for features $B_{pre}/B_{com}/B_{sem}$ to adaptively weight the feature channels. The weighted features are then summed and passed through a 1×1 convolution and CCA attention to obtain the fused features F_{SCB}. The fused features can be expressed as:

$$\begin{aligned} F_{SCB} = \Phi \{ & \lambda \left[N(B_{pre}) \right] \times B_{pre} \\ & + \lambda \left[N(B_{com}) \right] \times B_{com} \\ & + \lambda \left[N(B_{sem}) \right] \times B_{sem} \end{aligned} \tag{4.4}$$

where λ denotes the sigmod function. Φ is the 1×1 convolution. The B_{pre} represents features from the previous stage.

***LBSCNet Total Loss Function*:** We train the whole network end-to-end. The multi-task loss L_{total} is expressed as :

$$L_{total} = 3 \times L_{bev} + L_{sem} + L_{com} \tag{4.5}$$

where L_{bev}, L_{sem} and L_{com} respectively represent BEV loss, the semantic loss and completion loss.

4.4 Aerial-Ground Motion Planning

In this section, we introduce the novel AG-Planner. It is built on EGO-Planner [75] and consists of **1)** an energy-efficient Kinodynamic A* path searching front-end, **2)** a gradient-based trajectory optimization back-end and **3)** a post-refinement procedure. Our AG-Planner evaluates and projects gradient information directly from obstacles

Algorithm 1: Energy-Efficient Kinodynamic A* Search

Input: Start State x_s and Target State x_g
Output: Energy-Efficient Valid Path between x_s and x_g
Data: O=$\emptyset$ and C= $\emptyset$; $f(x_s) = g(x_s) + h(x_s)$; $O.push(x_s)$

while *!O.empty()* **do**
  $x \longleftarrow O.popMin()$
  if $x == x_g$ **then**
    return *path*
  end
  else
    $C.push(x)$
    foreach $n \in neig(x)$ **do**
      $g_n \longleftarrow (um.squaredNorm() + w_{time}) * \tau + g(x)$
      `// next node flying`
      if $z \geq ground\_judge$ **then**
        $g_n -= x.fly\_penalty\_g$
        `// add fly penalty cost`
        $g_n += fly\_cost * z + f\_cost\_base$
        $fly\_penalty\_g = fly\_cost * z + f\_cost\_base$
        $steer\_penalty\_g = 0$
        $next\_motion\_state = true$
      end
      `// next node driving`
      else
        $g_n -= x.steer\_penalty\_g$
        `// add steer penalty cost`
        $steer\_cost = steer\_cost\_ * pow(\omega_z, 2)$
        $g_n += steer\_cost + ground\_cost\_base$
        $steer\_penalty\_g = steer\_cost + g\_cost\_base$
        $fly\_penalty\_g = 0$
        $next\_motion\_state = false$
      end
      $f_n = g_n + \lambda * estimateHeuristic(n, x_g)$
      if $n \notin O \cup C$ **then**
        $n.updateCost(g_n, fly\_penalty, steer\_penalty, f_n)$
        $O.push(n)$
      end
    end
  end
end
return *null* `// Cannot find a valid path`

instead of a pre-built ESDF like [68]. To the best of our knowledge, AG-Planner is the first ESDF-free and energy-efficient planner tailored for AGRs.

4.4.1 Energy-Efficient Kinodynamic Hybrid A* Path Searching

Our AG-Planner first creates a naive "*initial trajectory*" ι (in Fig. 4.4a) that overlooks obstacles by randomly adding coordinate points, considering the positions of both the starting and target points. Following that, for the "*collision trajectory segment*" (i.e., the trajectory inside the obstacle), the back end of our planner based on [11] to propose an energy-efficient kinodynamic A* path search algorithm (in Alg. 1) to establish a safe "*guidance trajectory segment*" τ, which uses motion primitives instead of straight lines as graph edges in the searching loop. In this algorithm, we add extra flying and ground-steering energy consumption for the motion primitives (in Fig. 4.4a). Consequently, the

path searching not only tends to plan ground trajectories and avoid large turns but also switches to aerial mode and flies over them only when AGRs encounter huge obstacles, thereby promoting energy-saving.

4.4.2 Gradient-Based B-spline Trajectory Optimization

***B-spline Trajectory Formulation*:** In trajectory optimization (in Fig. 4.4b), the trajectory is parameterized by a uniform B-spline curve Θ, which is uniquely determined by its degree p_b, N_c control points $\{Q_1, Q_2, Q_3, ..., Q_{N_c}\}$, and a knot vector $\{t_1, t_2, t_3, ..., t_{M-1}, t_M\}$, where $Q_i \in \mathbb{R}^3, t_m \in \mathbb{R}, M = N + p_b$. Following the matrix representation of the [4] the value of a B-spline can be evaluated as:

$$\Theta(u) = [1, u, ..., u^p] \cdot M_{p_b+1} \cdot [Q_{i-p_b}, Q_{i-p_b+1}, ..., Q_i]^T \tag{4.6}$$

where M_{p_b+1} is a constant matrix depends only on p_b. And $u = (t - t_i)/(t_{i+1} - t_i)$, for $t \in [t_i, t_{i+1})$.

In particular, in ground mode, we assume that AGR is driving on flat ground so that the vertical motion can be omitted and we only need to consider the control points in the two-dimensional horizontal plane, denoted as $Q_{ground} = \{Q_{t0}, Q_{t1}, ..., Q_{tM}\}$, where $Q_{ti} = (x_{ti}, y_{ti}), i \in [0, M]$. In aerial mode, the control points are denoted as Q_{aerial}. According to the properties of B-spline: the k^{th} derivative of a B-spline is still a B-spline with order $p_{b,k} = p_b - k$, since $\triangle t$ is identical alone Θ, the control points of the velocity V_i, acceleration A_i and jerk J_i curves are obtained by:

$$V_i = \frac{Q_{i+1} - Q_i}{\triangle t}, A_i = \frac{V_{i+1} - V_i}{\triangle t}, J_i = \frac{A_{i+1} - A_i}{\triangle t} \tag{4.7}$$

***Collision Avoidance Force Estimation*:** Inspired by [75], for each control point on the collision trajectory segment, vector v (i.e., a safe direction pointing from inside to outside of that obstacle) is generated from ι to τ and p is defined at the obstacle surface (in Fig. 4.4a). With generated $\{p, v\}$ pairs, the planner maximizes D_{ij} and returns an optimized trajectory. The obstacle distance D_{ij} if i^{th} control point Q_i to j^{th} obstacle is defined as:

$$D_{ij} = (Q_i - p_{ij}) \times v_{ij} \tag{4.8}$$

Because the guide path τ is energy-saving, the generated path is also energy efficient (in Fig. 4.4a).

***B-spline Trajectory Optimization and Post-refinement Procedure*:** The basic requirements of the B-spline paths are three-fold: *smoothness*, *safety*, and *dynamical feasibility*. Based on the special properties of AGR bimodal, we first adopt the following cost terms designed by *Zhou et al.* [75]:

$$\min J_1 = \lambda_s J_s + \lambda_c J_c + \lambda_f (J_v + J_a + J_j) \tag{4.9}$$

where J_s is the smoothness penalty, J_c is for collision, and J_v, J_a, J_j are dynamical feasibility costs that limit velocity, acceleration and jerk. $\lambda_s, \lambda_c, \lambda_f$ are weights for each

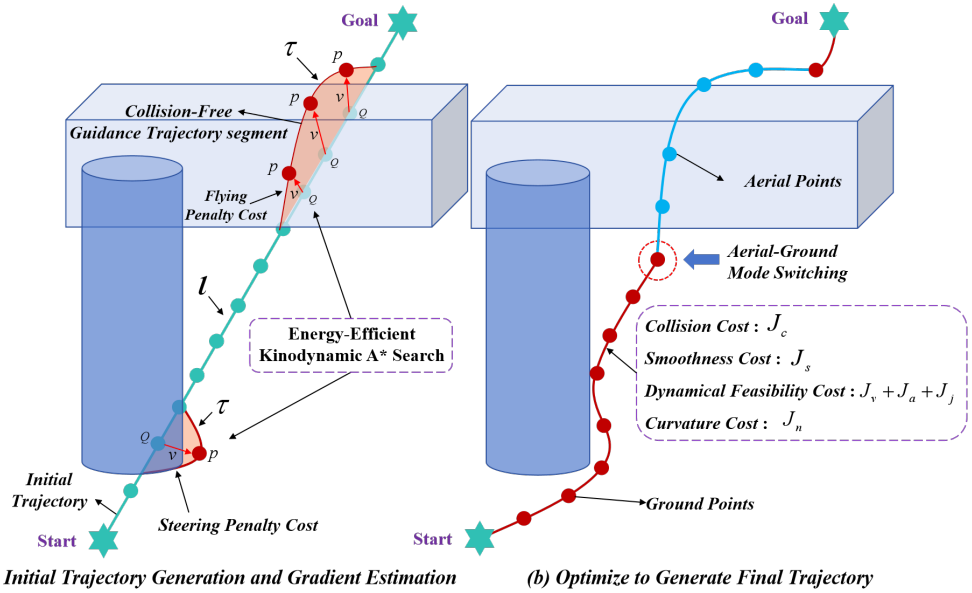


Figure 4.4: Illustration of AG-Planner and topological trajectory generation.

cost terms. Detailed explanations can be found in [75]. Subsequently, based on our observations, AGR faces non-holonomic constraints when driving on the ground, which means that the ground velocity vector of AGR must be aligned with its yaw angle. Additionally, AGR needs to deal with curvature limitations that arise due to minimizing tracking errors during sharp turns. Therefore, a cost for curvature needs to be added, and J_n can be formulated as:

$$J_n = \sum_{i=1}^{M-1} F_n(Q_{ti}) \tag{4.10}$$

where $F_n(Q_{ti})$ is a differentiable cost function with C_{max} specifying the curvature threshold:

$$F_n(Q_{ti}) = \begin{cases} (C_i - C_{max})^2, C_i > C_{max}, \\ 0, C_i \leq C_{max} \end{cases} \tag{4.11}$$

where $C_i = \frac{\triangle \beta_i}{\triangle Q_{ti}}$ is the curvature at Q_{ti}, and the $\triangle \beta_i = \left| \tan^{-1} \frac{\triangle y_{ti+1}}{\triangle x_{ti+1}} - \tan^{-1} \frac{\triangle y_{ti}}{\triangle x_{ti}} \right|$. In general, the overall objective function is formulated as follows:

$$\begin{aligned} &\min J_{all} = \lambda_s J_s + \lambda_c J_c + \lambda_f (J_v + J_a + J_j) + \lambda_n J_n \\ &s.t. \begin{cases} J_s = \sum_{i=1}^{N_c-1} \|A_i\|_2^2 + \sum_{i=1}^{N_c-2} \|J_i\|_2^2 \\ J_c = \sum_{i=1}^{N_c} j_c(Q_i) \\ J_v = \sum_{i=1}^{N_c} \omega_v F(V_i) \\ J_a = \sum_{i=1}^{N_c-1} \omega_a F(A_i) \\ J_j = \sum_{i=1}^{N_c-2} \omega_j F(J_i) \\ J_n = \sum_{i=1}^{M-1} F_n(Q_{ti}) \end{cases} \end{aligned} \tag{4.12}$$

The optimization problem is solved using the non-linear optimization solver NLopt [22], with post-refinement from [75] for constraint violations. After path planning, a setpoint from the trajectory is selected and sent to the controller. Aerial setpoints include yaw angle and 3D position, velocity, and acceleration, while ground ones include yaw angle and 2D position and velocity. In addition, when the z-axis coordinate of the next control point is greater than the ground threshold, that is, when mode switching is required, an additional trigger signal will be sent to the controller (i.e., PX4 Autopilot). The controller will automatically switch to the flight state.

4.5 Evaluation

In this section, we first assess the LBSCNet-based perception module on the SemanticKITTI benchmark, examining its accuracy and rapid inference capabilities in SSC tasks. Subsequently, we integrate this module with the AG-Planner by deploying a pre-trained model offline, forming a comprehensive HE-Nav system. We then evaluate the AGR's autonomous navigation capability using HE-Nav in both simulated and real-world settings, focusing on ***performance*** metrics (i.e., planning success rate, average movement time) and ***efficiency*** aspects (i.e., average planning time, energy consumption).

4.5.1 Evaluation setup

Perception Module: For training and testing of LBSCNet, we utilized a server with 4 NVIDIA RTX 3090 GPUs and 128GB memory, employing the outdoor SemanticKITTI dataset [15]. We trained the model for 80 epochs on a single NVIDIA 3090 GPU with a batch size of 12, using the Adam optimizer [27] at an initial learning rate of 0.001, and augmenting the input point cloud by random flipping along the $x-y$ axis. Ultimately, we deployed the pre-trained model offline with the best completion accuracy to complete the local map.

Simulation Experiment: Experiments were executed on a laptop equipped with Ubuntu 20.04, an i9-13900HX CPU, and an NVIDIA RTX 4060 GPU to simulate aerial-ground robotic navigation within complex environments. The test scenarios comprised a $20m \times 20m \times 5m$ square room and a $3m \times 30m \times 5m$ corridor with numerous random obstacles, creating occluded spaces and unknown areas (Fig. 4.9A). The AGR's task was to navigate from a starting point to a designated destination without collision.

Indoor and Outdoor Real-world Experiment: We employed HE-Nav on a custom AGR platform (Fig. 4.5) for indoor and outdoor experiments, using Prometheus software [1] with a RealSense D435i depth camera, a T265 tracking camera, and a Jetson Xavier NX computer. Hardware details are in the supplementary materials. We assessed the average energy consumption per second for AGR during driving and flying (Table 4.4) to establish a basis for evaluating energy usage in real and simulated tests.

Metrics: For the LBSCNet, we use intersection over union (IoU) to evaluate scene completion quality and the mean IoU (mIoU) of 19 semantic classes to assess semantic segmentation performance. moreover, we also focus on LBSCNet's inference speed to ensure it meets the real-time requirements for autonomous navigation. Regarding

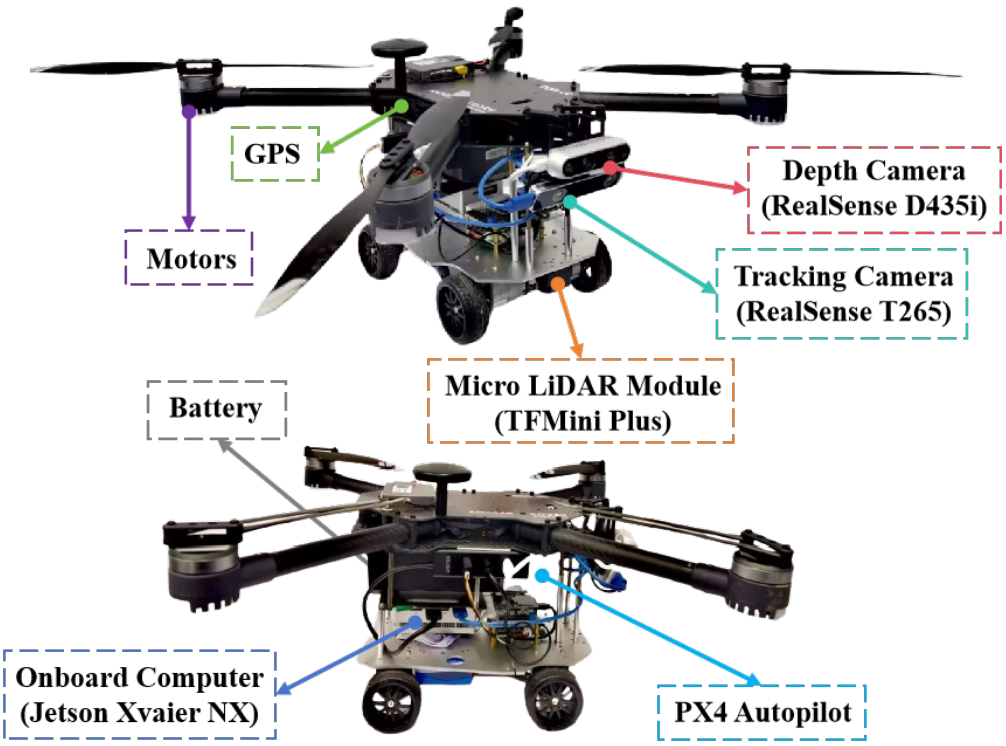


Figure 4.5: The detailed composition of the robot platform.

planning, we pay attention to performance metrics such as planning success rate (%), total moving time (s), and efficient metrics planning time (ms) as well as energy consumption (J).

***Baseline methods*:** For the perception module, we compare LBSCNet against the state-of-the-art SSC methods: (1) Camera-based SSC method MonoScene [5] and VoxFormer[29], (2) Point-Cloud-based SSC methods including LMSCNet [43], and SSCNet [47] and SCPNet[58]. To evaluate the performance and efficiency of HE-Nav, we compared HE-Nav with two state-of-the-art AGRs navigation systems: TABV [68], HDF [13].

4.5.2 LBSCNet Comparison against the state-of-the-art.

Quantitative Results: We evaluated our proposed LBSCNet against state-of-the-art SSC methods on the SemanticKITTI test datasets by submitting results to the official test server. Table 4.1 demonstrates that LBSCNet not only achieves the highest completion metric IoU (59.71%) but also ranks third in the semantic segmentation metric mIoU (23.58%). Although SCPNet's semantic segmentation accuracy surpasses ours, its dense network design renders it incapable of real-time operation (i.e., FPS < 1). In contrast, LBSCNet outperforms SCPNet by 6.43% in IoU and runs approximately ***20 times*** faster in a single RTX 3090 GPU.

The remarkable accuracy and rapid inference performance of our LBSCNet are primarily due to the innovative semantic and completion decoupling network structure, which exploits contextual semantic information to bolster scene understanding and

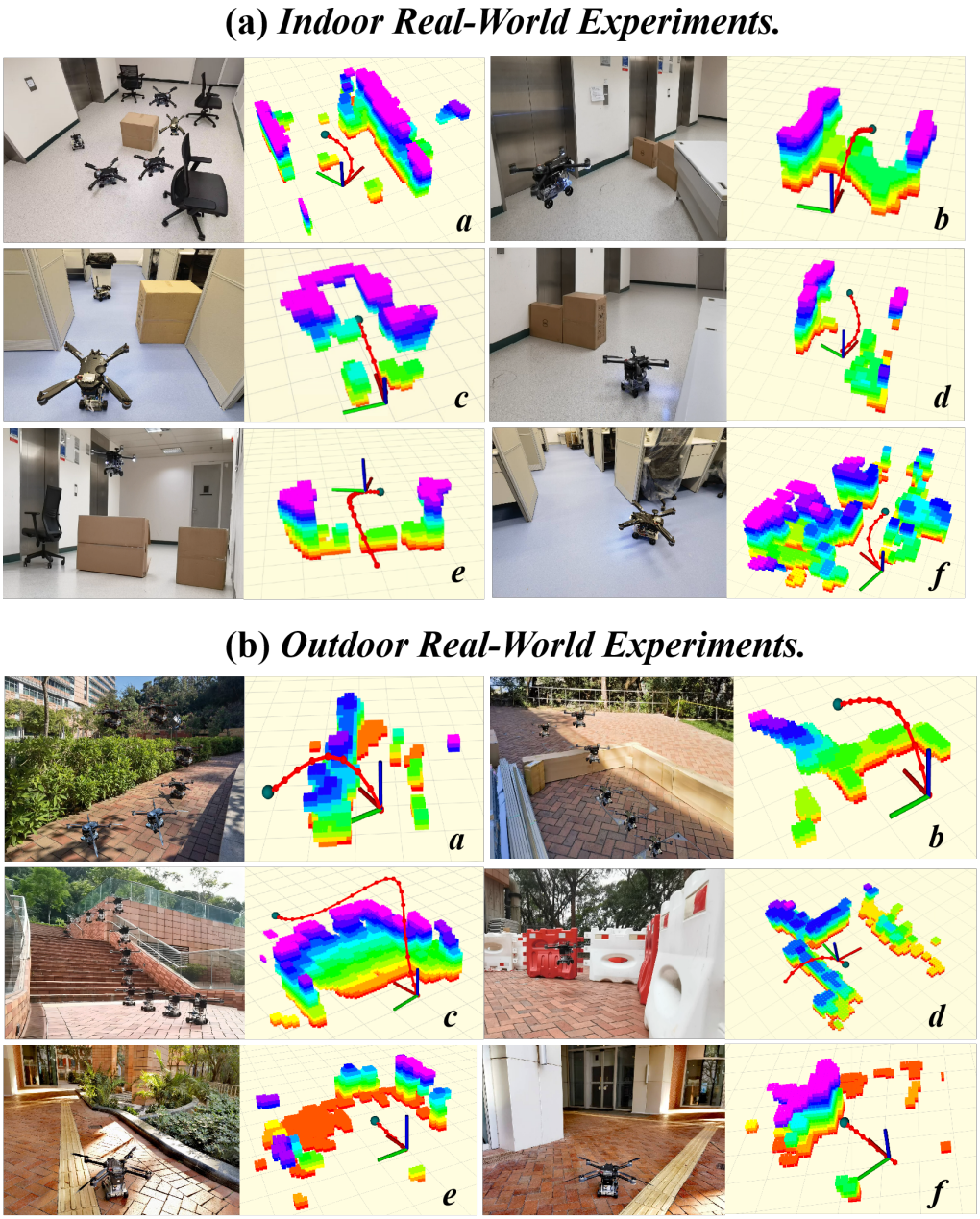


Figure 4.6: HE-Nav's visual results showcase its autonomous navigation capabilities in 6 indoor and 6 outdoor scenes. The system effectively predicts obstacle distribution in occluded areas and plans collision-free hybrid trajectories.

Table 4.1: Quantitative comparison against the state-of-the-art SSC methods on the official SemanticKITTI test benchmark.

Method	*IoU*	*mIoU*	*Prec.*	*Recall*	*FPS*
SSCNet [47]	53.20	14.55	59.13	**84.15**	12.00
LMSCNet [43]	55.32	17.01	77.11	66.19	13.50
LMSCNet-SS [43]	56.72	17.62	**81.55**	65.07	13.50
S3CNet [7]	45.60	29.50	48.79	77.13	1.20
Monoscene [5]	38.55	12.22	51.96	59.91	< 1
VoxFromer-T [29]	57.69	18.42	69.95	76.70	< 1
VoxFromer-S [29]	57.54	16.48	70.85	75.39	< 1
SCPNet [58]	56.10	**36.70**	72.43	78.61	< 1
LBSCNet (Ours)	**59.71**	23.58	77.60	71.29	**20.08**

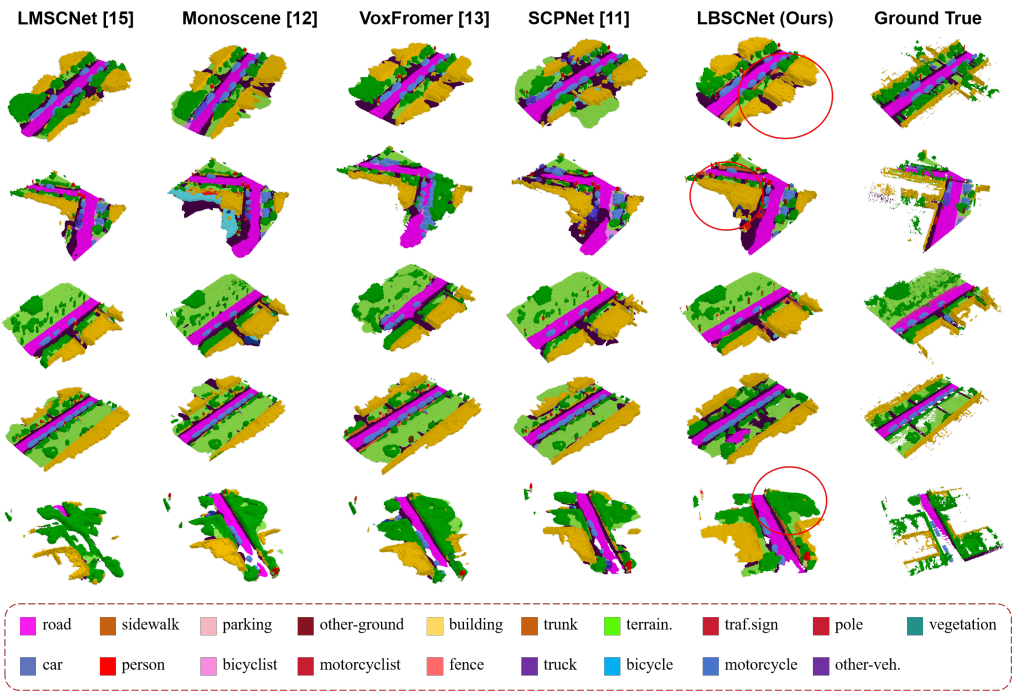


Figure 4.7: Qualitative results of our method and others. LBSCNet better captures the scene layout in large-scale scenarios.

completion. The integration of the novel SCB-fusion and CCA modules enables the network to remain lightweight while significantly enhancing completion accuracy by capturing contextual features and learning long-distance dependencies. Additionally, the employment of sparse 3D convolutions and lightweight BEV feature fusion ensures low latency and high-speed inference (20.08 FPS), making LBSCNet ideal for real-time perception in AGR navigation systems. Further details can be found in Table 4.5 and Fig 4.11.

Qualitative Results: We provide visualizations results on the SemanticKITTI validation set and include results from LMSCNet [43], Monoscene[5], VoxFormer[29], and SCPNet [58]. As illustrated in Fig. 4.7, our LBSCNet demonstrates superior SSC predictions, particularly for “wall" classes and larger objects like cars, aligning with the results in Table 4.1. Importantly, the occlusion areas we target, such as vegetation and trees behind walls, are accurately completed, proving vital for subsequent path-planning applications. More qualitative and quantitative results are provided in the supplementary material, i.e., in Section 4.7.1.

Table 4.2: Ablation study of our model design choices on the SemanticKITTI validation set.

Method	IoU ↑	mIoU ↑
LBSCNet (ours)	58.34	22.74
w/o SCB-Fusion Module	57.05	21.26
w/o Criss-Cross Attention	57.20	22.17

Ablation Study: Ablation studies conducted on the SemanticKITTI validation set (Table 4.2) emphasize the significance of two crucial components in our network: CCA attention mechanisms and the SCB-Fusion Module. The CCA attention mechanism greatly influences completion accuracy by effectively aggregating context across rows and columns. The absence of CCA results in a 1.95% decrease in completion accuracy. On the other hand, the SCB-Fusion module captures local scene features, including occluded areas, with minimal computational overhead. Removing the SCB-Fusion module leads to a 2.21% reduction in IoU.

4.5.3 Simulated Air-Ground Robot Navigation

In a square room and corridor scenario (Fig 4.9B), we conducted a comparative analysis of our HE-Nav system, TABV [68], and HDF [13]. Through 100 trials with varied obstacle placements, we evaluated the average moving time, planning time (including updating the ESDF map and path planning for TABV), and success rate (i.e., collision-free) of each system (Fig. 4.8). Furthermore, we obtained average energy consumption results for the 100 simulated trials by combining recorded flight and driving times with real-world energy consumption data from our custom AGR (Table 4.4).

The results shown in Fig. 4.8 highlight the exceptional performance of our HE-Nav system. It achieves high success rates of 98% and 97% in square rooms and corridors, respectively, with average movement times of 12.2s and 16.2s. The average planning

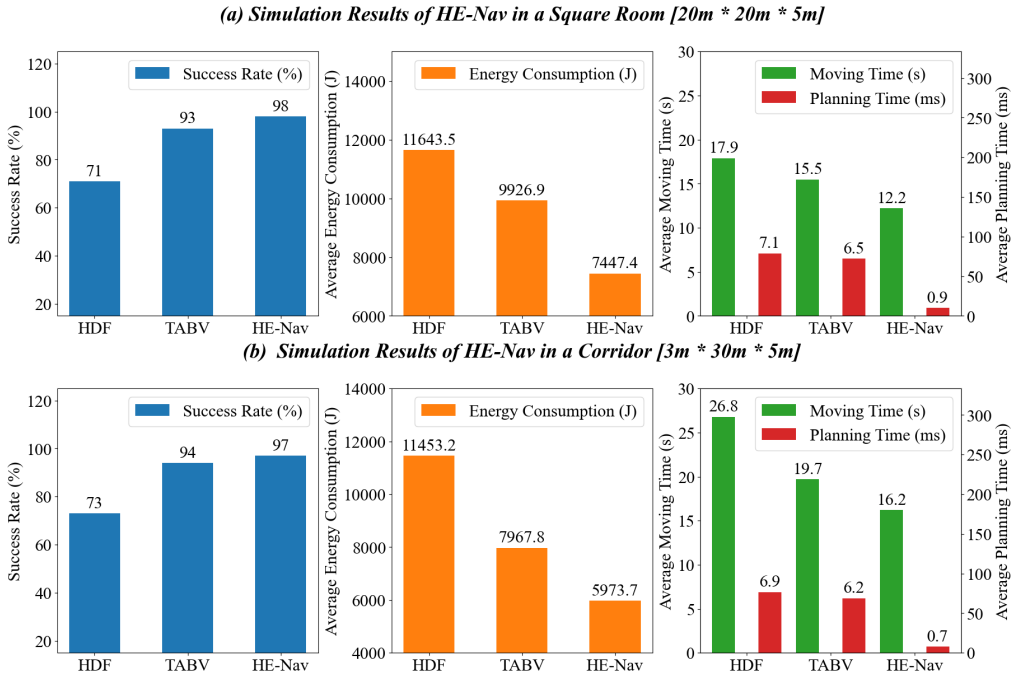
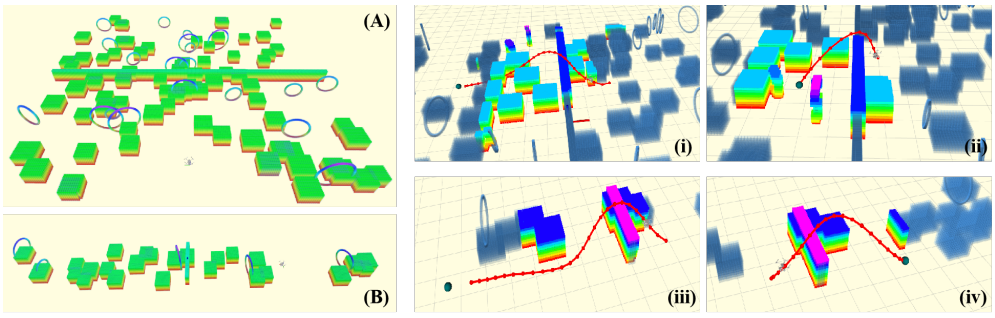


Figure 4.8: Quantitative results of HE-Nav in two simulation scenarios.



(a) *Two Simulation Scenarios* (b) *HE-Nav Navigation results*

Figure 4.9: Qualitative results of path planning and occlusion prediction in simulation environment.

time is significantly accelerated, being 6 times faster than TABV [68] and HDF [13], thanks to the elimination of redundant ESDF calculations. Additionally, our path planner seamlessly integrates with the *energy-efficient Kinodynamic A* algorithm*, resulting in the lowest average energy consumption of 7447.4 J and 5973.7 J. Comparatively, HE-Nav achieves a 24.98% energy reduction in square scenarios and a 25.03% reduction in corridors compared to TABV [68].

In contrast to TABV [68], which primarily focuses on flight energy consumption and lacks the ability to sense obstacle distribution in occluded areas beforehand, our HE-Nav system addresses this limitation effectively. By perceiving and predicting occlusions, our ESDF-free AG-Planner can bypass these regions and significantly reduce collision risks. This not only results in more optimal overall energy consumption but also greatly mitigates the risk of collision for the planned path.

Methods	**Indoor Scenario**						**Outdoor Scenario**					
	a	b	c	d	e	f	a	b	c	d	e	f
TABV [68]	3217.4	10207.3	3971.8	5783.5	12362.9	3105.6	10323.6	10569.4	13117.2	12649.3	6480.5	6682.8
HE-Nav (Ours)	2891.7	9652.5	3614.1	5279.3	11874.5	2765.9	9662.9	9764.8	11283.2	11858.7	5579.3	5754.2
HE-Nav Energy Savings	10.12%	5.43%	9.01%	8.71%	3.95%	10.93%	6.41%	7.61%	13.95%	6.25%	13.89%	13.90%
Average Energy Savings				8.02%						10.34%		

Table 4.3: Quantitative results of AGR energy consumption (J) in complex indoor and outdoor scenes.

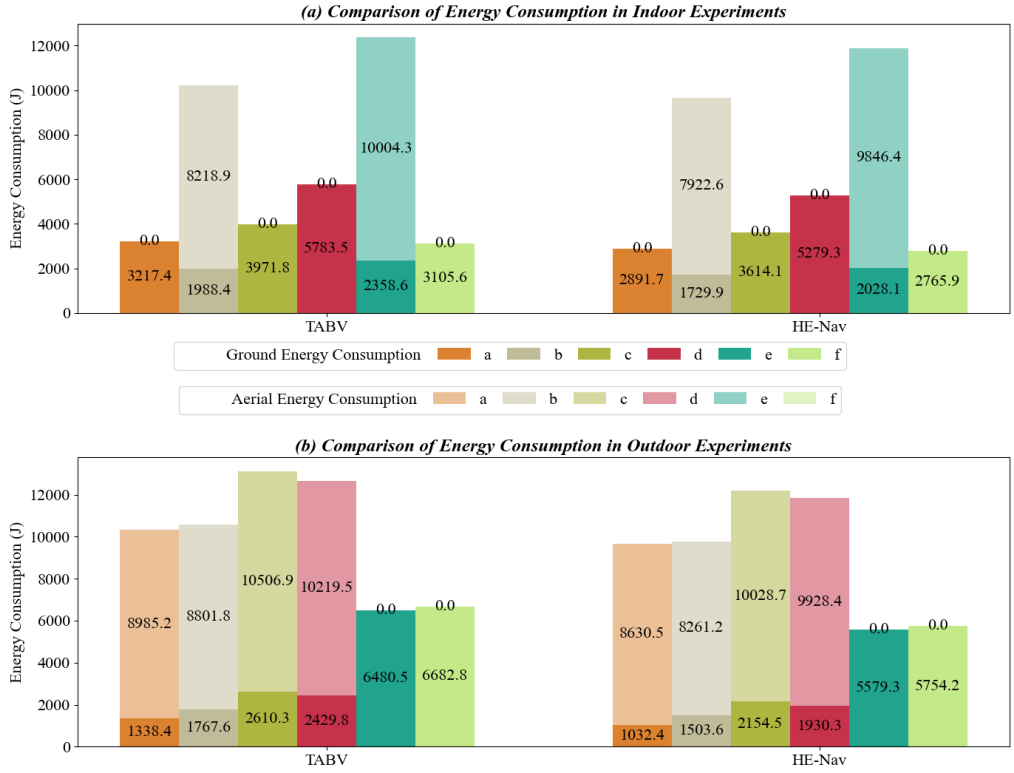


Figure 4.10: Quantitative results of indoor and outdoor real environmental energy consumption.

4.5.4 Real-world Air-Ground Robot Navigation

We assess HE-Nav's performance and energy efficiency across 6 indoor and 6 outdoor scenarios (Fig. 4.6). In indoor settings, as illustrated in Tab. 4.3 and Fig. 4.10a, HE-Nav consistently demonstrates lower average energy consumption than TABV [68]. For example, in scenarios a, c, and f, our system achieves ground energy consumption reductions of 10.12%, 9.01%, and 10.93% compared to TABV, primarily attributable to the incorporation of additional turning penalty terms in the ground segment. This approach effectively minimizes energy usage in ground mode by reducing high-angle turning paths. Concurrently, LBSCNet swiftly predicts obstacle distribution in occluded areas, constructing a more complete local map (e.g., a, c, f visualization results) to serve as the foundation for AG-Planner's search path.

Transitioning to outdoor scenarios, HE-Nav surpasses TABV [68] with a 10.34% reduction in average energy consumption (Table 4.3), mainly due to the optimization of smooth aerial paths, which minimizes flight energy consumption. Our system adeptly predicts obstacle distribution in cluttered and occluded environments (i.e., a, b, c), while fully accommodating AGR's non-holonomic constraints and energy costs. In ground mode, the planned path exhibits smoothness and dynamic feasibility (e.g., b, e, f visualization results). Crucially, scenes e and f display substantial reductions in ground energy consumption of 13.89% and 13.90%, respectively, owing to challenging terrain (e.g., the yellow blind road in e, f). In contrast to the TABV method, which neglects ground steering energy consumption, HE-Nav integrates ground steering constraints into the optimization process, thereby enhancing energy efficiency on difficult terrains.

Furthermore, the removal of ESDF significantly reduced the overall path planning time, achieving an approximate 8x improvement compared to the ESDF-based TABV (Fig. 4.15). The average planning time, including ESDF updating, was reduced to 0.95ms and 1.12ms on the Jetson Xavier NX platform. For additional qualitative and quantitative results, please refer to the supplementary material and video, specifically Section 4.7.3.

4.6 Conclusion

we have presented HE-Nav, the first high-performance, efficient and ESDF-free navigation system specifically designed for aerial-ground robots (AGRs). By integrating innovative features such as the lightweight BEV-guided semantic scene completion network (LBSCNet) and the aerial-ground motion planner (AG-planner), our system is capable of predicting obstacle distributions in occluded areas and generating low-collision risk, energy-efficient aerial-ground hybrid trajectories in real-time (≈ 1 ms). Through extensive simulations and real experiments, HE-Nav has been shown to significantly outperform recent planning frameworks in performance (i.e., planning success rate and total movement time) and efficiency (i.e., planning time and energy consumption).

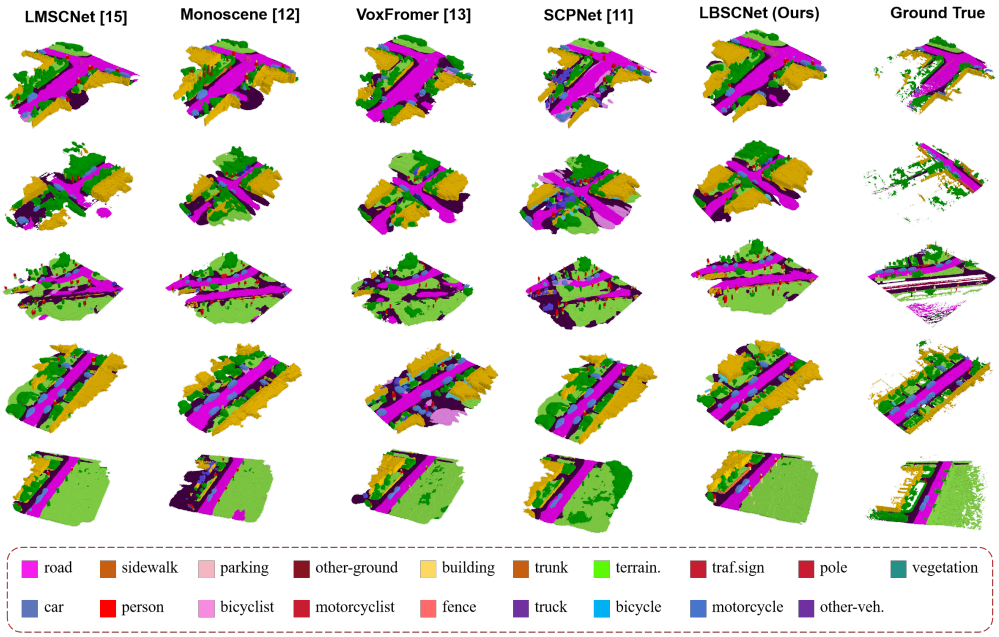


Figure 4.11: Qualitative results of our method and others. LBSCNet better captures the scene layout in large-scale scenarios.

4.7 Supplementary Section

In the supplementary Section, we discuss additional implementation details and provide more qualitative and quantitative results about *LBSCNet*, *Simulation Experiments* and *Real-World Experiments*.

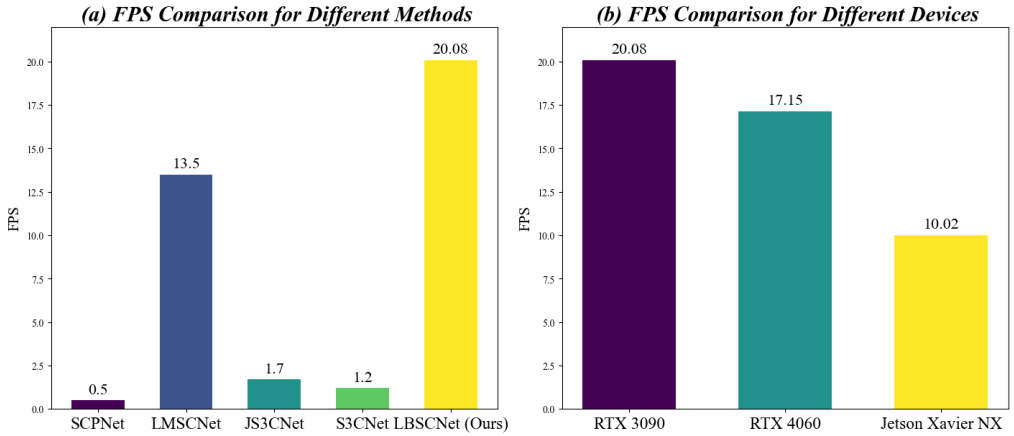
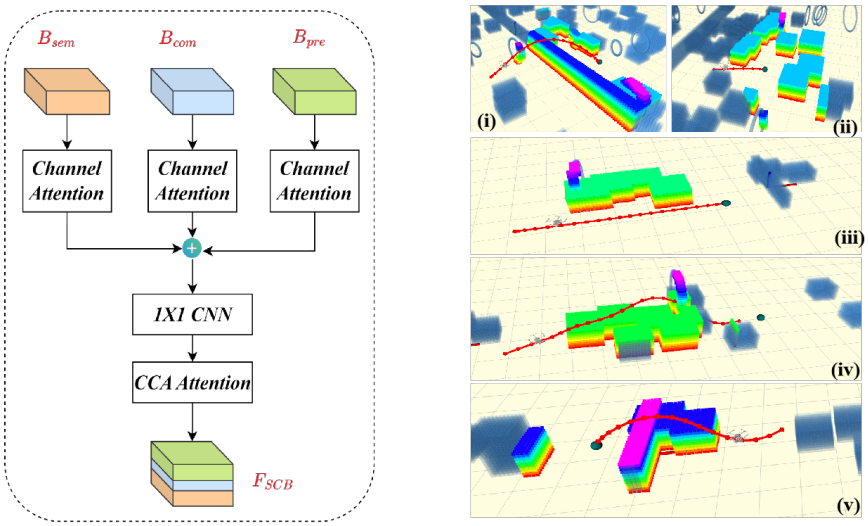


Figure 4.12: Qualitative results of our method and others. LBSCNet better captures the scene layout in large-scale scenarios.

4.7.1 LBSCNet

In Table 4.5, we present an extensive array of quantitative results, encompassing completion accuracy and semantic segmentation accuracy. Moreover, the visualization of outcomes in the SemantiKITTI dataset validation set is depicted in Fig. 4.11. It is evident that LBSCNet excels in comparison to other methods with respect to completion

and semantic representation of roads, vehicles, buildings, and vegetation, which is in alignment with the findings displayed in Table 3. Despite our semantic segmentation results ranking third among all approaches, we possess superior completion accuracy and real-time performance. This is of paramount importance for Autonomous Ground Robots (AGRs) to accurately and promptly predict the distribution of obstacles in occluded areas during navigation.



(a) Our SCB-Fusion module structure. **(b) Qualitative results in the simulation environment.**

Figure 4.13: SCB-Fusion Module and Qualitative results in the simulation environment.

Parameter	**Value**
Battery Capacity	10000 mAh
Battery Weight	1008 g
Rated Power	231 Wh
Operating Voltage	23.05 V
Driving Energy Consumption	$\approx$ 251.45 J/s
Flying Energy Consumption	$\approx$ 988.33 J/s

Table 4.4: Battery and Energy Consumption Parameters

In addition, as illustrated in Fig. 4.12a, the inference speed comparison of LBSCNet highlights its performance advantages. Owing to their dense 3D convolution design, existing point-cloud-based SSC methods are unable to achieve real-time inference. Concurrently, Fig. 4.12b demonstrates the inference speed of LBSCNet on various devices. It achieves 20.08 FPS on an RTX 3090 GPU and 17.15 FPS on an RTX 4060 GPU (i.e., in a simulated experiment). Furthermore, when optimized by TensorRT on a Jetson Xavier NX, LBSCNet attains a real-time performance of 10.02 FPS (i.e., in a real-world experiment).

Method	LBSCNet	SCPNet	VoxFormer	MonoScene	LMSCNet
IoU (%)	**59.71**	56.10	57.69	38.55	54.89
Precision (%)	78.60	68.13	69.95	51.96	**82.21**
Recall (%)	71.29	74.92	**76.70**	59.91	62.29
mIoU (%)	23.58	**36.70**	18.42	12.22	14.13
car	35.80	**46.40**	37.46	24.64	35.41
bicycle	8.00	**33.20**	2.87	0.23	0.00
motorcycle	4.10	**34.90**	1.24	0.20	0.00
truck	4.90	13.80	**10.38**	13.84	3.49
other-veh.	8.10	**29.10**	10.61	2.13	0.00
person	3.40	**28.20**	3.50	1.37	0.00
bicyclist	2.70	**24.70**	3.92	1.00	0.00
motorcyclist	1.80	1.80	0.00	0.00	0.00
road	**71.30**	68.50	66.15	57.11	67.56
parking	39.40	**51.30**	23.96	18.60	13.22
sidewalk	42.90	**49.80**	34.53	27.58	34.20
other-grnd	16.70	**30.70**	0.76	2.00	0.00
building	**43.40**	38.80	29.45	15.97	27.83
fence	31.50	**44.70**	11.15	7.37	4.42
vegetation	45.10	**46.40**	38.07	19.68	33.32
trunk	26.20	**40.10**	12.75	2.57	3.01
terrain	40.90	**48.70**	39.61	31.59	41.51
pole	15.00	**40.40**	15.56	3.79	4.43
traf.-sign	6.80	**25.10**	8.09	2.54	0.00

Table 4.5: Quantitative comparison against the state-of-the-art SSC methods.

4.7.2 Simulation Experiment

More qualitative simulation results are shown in Fig. 4.13b. The prediction results are updated to the local map with low latency, which allows AG-Planner to avoid these areas in advance when searching for paths.

4.7.3 Real-World Experiment

For autonomous navigation, we equip the AGR with the following onboard devices:

- **RealSense D430 depth camera :** This camera provides the depth point clouds for local map fusion. Point clouds are also input to the LBSCNet network.
- **RealSense T265 tracking camera :** This camera provides robust Visual Inertial Odometry (VIO) for UAV state estimation.
- **PX4 Autopilot :** It provides onboard IMU measurements and serves as the inner-loop controller.
- **TF Mini Plus :** It provides robust height information for the robot because T265 will drift outdoors, especially in the Z-axis direction.
- **Jetson Xavier NX :** It is an onboard computer with a 6-core NVIDIA CPU and 8 GB RAM. The entire HE-Nav, including map fusion, state estimation, motion planning and control modules, runs on it.

The battery information used by our customized AGR is shown in Table 4.4. We tested the energy consumption in the real environment to obtain the corresponding

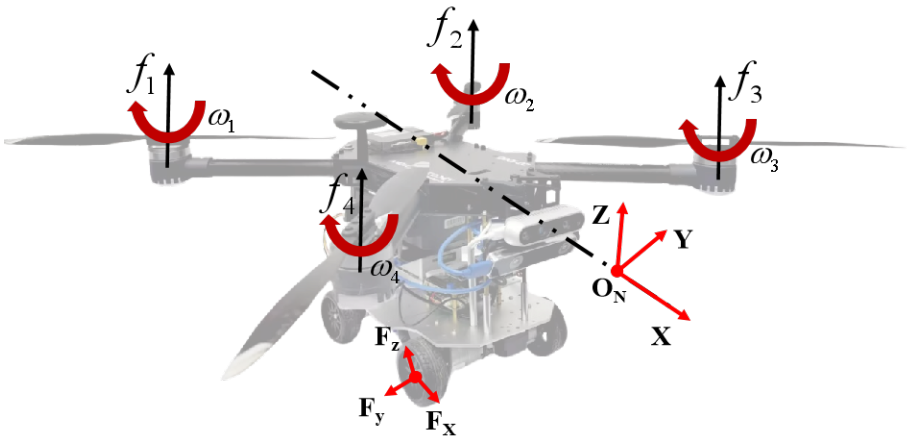


Figure 4.14: An equivalent mathematical model for the AGR.

energy consumption based on the driving time and flight time recorded in the simulation experiment and the real environment experiment. The relative pose relationship of the AGRs platform was represented by a body coordinate system $(B-xyz)$, and a global coordinate system $(O-XYZ)$, Fig. 4.14. These were used to establish additional coordinate systems, the body's angular velocity in the body coordinate system $\omega_n = [\phi, \theta, \psi]^T$, and angular velocity in global coordinate system $\omega_b = [p, q, r]^T$. where ϕ, θ, ψ represent the roll, pitch, and yaw angles for the body relative to the global coordinate system. The terms p, q, r denote the roll, pitch, and yaw angular velocities for the body relative to the body coordinates, respectively. Furthermore, our customized AGR can continuously fly for 14 minutes or drive on the ground for 55 minutes using a 10000mAh power source. When combined with our HE-Nav navigation system, its energy efficiency is maximized, making it well-suited for continuous operation in complex and occluded wild environments. We encourage reviewers to view our supplementary video for additional information.

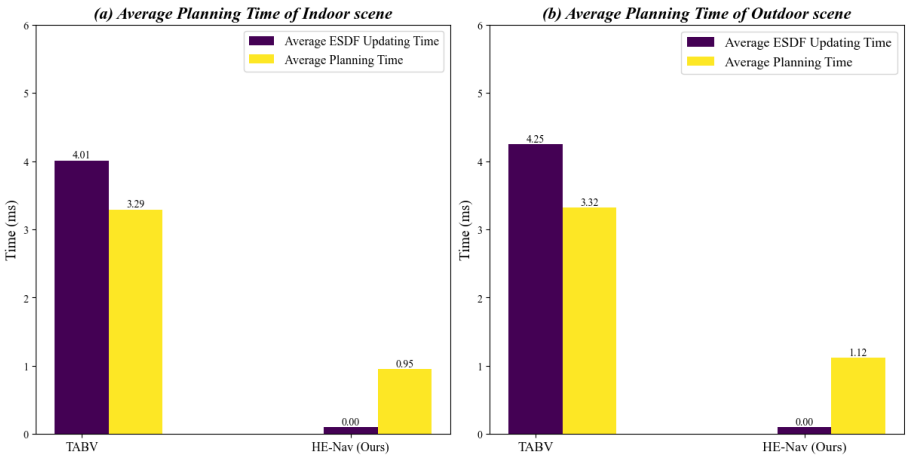


Figure 4.15: Total planning time of HE-Nav on Jetson Xavier NX (i.e. ESDF updating time + planning time).

Chapter 5

Conclusion and Future Work

This thesis introduces the development of two innovative navigation systems for air-ground robots, aimed at enhancing performance and optimizing energy consumption. The focus of our work is to address and ameliorate the limitations identified in prior research within the domains of perception networks and path-planning algorithms. Specifically, existing perception networks suffer from high memory demands and an inability to operate in real-time, whereas their lightweight counterparts struggle with capturing contextual nuances. Additionally, earlier path planning strategies, particularly those based on the Euclidean Signed Distance Field (ESDF), failed to deliver millisecond-level path planning and were marred by inefficiencies in ESDF map generation. In response to these challenges, we have engineered two comprehensive navigation systems. By meticulously redesigning the perception and planning modules, our work substantially elevates the autonomous navigation capabilities of land and air robots, ensuring remarkable improvements in both performance and energy efficiency amidst complex operational environments.

Our first contribution, AGRNav, introduces a groundbreaking navigation system specifically designed for air-ground robots (AGRs) navigating through occlusion-prone environments. This innovative system not only demonstrates a remarkable 98% success rate in safely guiding AGRs through areas laden with unseen obstacles but also stands out for its agility in updating the predictive results to the grid map, ensuring low-latency performance crucial for real-time navigation. At the heart of AGRNav lies the integration of a bespoke lightweight semantic scene completion network, SCONet, which revolutionizes the traditional approach to obstacle prediction and semantics. By employing depth-separable convolutions, SCONet dramatically reduces computational demands while maintaining high inference speeds, achieving a real-time performance of 20 frames per second. This enhancement in speed does not come at the cost of accuracy; on the contrary, SCONet sets a new benchmark in precision, as evidenced by its state-of-the-art IoU score of 56.12 on the SemanticKITTI benchmark. Furthermore, AGRNav's path-planning algorithm capitalizes on the predictions made by SCONet to significantly curtail unnecessary aerial manoeuvres, thereby slashing energy consumption by an impressive 50% compared to existing methods. This efficiency is achieved through a novel query-based method for occupancy updates, which prioritizes the modification of free voxels post-scanning, effectively minimizing latency and

optimizing the planning process. Additionally, AGRNav's hierarchical path planner, adept at navigating both aerial and ground terrains, leverages the semantic insights provided by SCONet to dynamically adjust the robot's speed in navigable areas like roads, further enhancing energy conservation.

Our second contribution, HE-Nav, is the first high-performance, efficient, and ESDF-free navigation system meticulously engineered for aerial-ground robots (AGRs), addressing the critical limitations of existing navigation systems in complex and occluded environments. At its core, HE-Nav revolutionizes the conventional navigation paradigm through the introduction of LBSCNet and AG-Planner, two pioneering components that synergize to deliver unparalleled navigational capabilities. LBSCNet, a lightweight yet profoundly accurate perception module, leverages sparse 3D convolutions and a novel architectural design to predict obstacle distributions in occluded areas with exceptional speed and precision. This breakthrough is particularly noteworthy as it ensures rapid inference, achieving an impressive 20.08 frames per second, while simultaneously setting a new benchmark in prediction accuracy with an IoU of 59.71 on the SemanticKITTI benchmark. The AG-Planner, on the other hand, epitomizes innovation in path planning. By eschewing the traditional reliance on ESDF maps, it introduces an ESDF-free approach that utilizes a novel energy-efficient Kinodynamic A* algorithm. This algorithm meticulously considers the energy costs associated with both aerial and ground movements, including the often-overlooked energy implications of ground steering. Consequently, AG-Planner not only significantly reduces planning times by an 8x factor compared to ESDF-based methods but also achieves substantial reductions in energy consumption—24.98% and 25.03% in simulated scenarios, and 10.34% in real-world outdoor environments. Such efficiencies are made possible by the planner's adeptness at generating initial trajectories that smartly navigate around obstacles, thereby minimizing unnecessary movements and optimizing energy use.

In rigorous evaluations conducted on the SemanticKITTI benchmark and through simulated and real-world tests, HE-Nav has demonstrably outperformed existing AGR navigation baselines, showcasing its superior performance, real-time planning capabilities, and energy efficiency. These achievements underscore the transformative potential of HE-Nav in the realm of autonomous navigation, particularly in applications involving search, exploration, and rescue tasks in challenging, occluded environments. Through the development of LBSCNet and AG-Planner, HE-Nav not only overcomes the limitations of existing systems but also opens new avenues for future research and development in the field of robotic navigation, heralding a new era of high-performance, efficient, and intelligent autonomous systems.

[Future Direction 1: Real-Time AGR Navigation System for Complex Static Environments with Dynamic Obstacles.] Recent advancements in autonomous navigation systems, particularly for air-ground robots (AGRs), have demonstrated remarkable proficiency in navigating through complex static environments. Notable frameworks such as AGRNav and HE-Nav have set benchmarks in efficiency, energy conservation, and high-performance navigation through leveraging innovative perception modules

and path-planning algorithms. These systems, however, primarily address challenges presented by static obstacles and may not fully encapsulate the complexities introduced by dynamic obstacles such as pedestrians, vehicles, animals, etc. Dynamic obstacles [63, 62, 59] pose unique challenges due to their unpredictable movements and the necessity for real-time adjustments in the navigation strategy. The capability to predict the future positions of moving obstacles and to adaptively replan the trajectory in real-time is crucial for ensuring safety and operational efficiency in dynamic environments. Furthermore, the computational demands of processing and predicting in such scenarios necessitate more lightweight and efficient prediction networks to maintain the real-time performance of AGR systems.

Objectives:

1. **Enhance Dynamic Obstacle Prediction:** Develop advanced algorithms capable of accurately predicting the motion trajectories of dynamic obstacles in real-time, integrating these predictions into the AGR's navigation system to anticipate and avoid collisions.
2. **Optimize Perception Networks:** Investigate and implement lightweight, efficient perception networks that can quickly process environmental data and predict dynamic obstacle movements with minimal latency, ensuring the real-time responsiveness of the AGR navigation system.
3. **Real-Time Adaptive Path Planning:** Design and develop a path planning module that can rapidly adjust to the changing environment based on dynamic obstacle predictions, ensuring safe and efficient navigation without significant pauses or recalculations.
4. **System Integration and Evaluation:** Integrate the enhanced dynamic obstacle prediction and optimized perception networks into a cohesive AGR navigation system. Evaluate the system's performance in various dynamic environments, focusing on its ability to navigate safely and efficiently in real-time.

[Future Direction 2: Trajectory Planner for AGRs on Uneven Terrain.] The deployment of air-ground robots (AGRs) in diverse applications, from exploration to search and rescue missions, often requires navigation over uneven terrain [60, 21, 55, 44]. This presents significant challenges for motion planning due to the need to assess terrain traversability and to adapt the robot's dynamics model to the complex terrain. Existing trajectory planning methods for uneven terrain often result in trajectories that are either too conservative, leading to inefficiencies, or not well-aligned with the robot's controller capabilities, leading to poor tracking performance. There's a pronounced need for a trajectory planner that can efficiently manage the energy consumption of AGRs while ensuring the dynamical feasibility and controller compatibility of the generated paths.

Objectives:

1. **Advanced Traversability Assessment:** Develop sophisticated algorithms for

precise assessment of uneven terrain traversability, incorporating real-time environmental data and robot-terrain interaction models.

2. **Dynamics Model Adaptation:** Design an adaptive dynamics model for AGRs that accurately reflects the impact of uneven terrain on robot motion, ensuring that the trajectory planning is both realistic and feasible.
3. **Energy-Efficient Trajectory Planning:** Create a trajectory optimization framework specifically for AGRs navigating uneven terrain, focusing on minimizing energy consumption without compromising the safety or dynamical constraints of the robot.
4. **Integration and Validation:** Integrate the developed components into a comprehensive navigation system for AGRs and validate its performance through simulations and real-world experiments, emphasizing the trajectory's energy efficiency and fidelity to the controller's tracking capabilities.

Bibliography

[1] Amovlab. *Prometheus UAV Open Source Project*. https://github.com/amov-lab/Prometheus.

[2] Jens Behley et al. "Semantickitti: A dataset for semantic scene understanding of lidar sequences". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 9297–9307.

[3] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. "The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 4413–4421.

[4] C de Boor. *Subroutine package for calculating with B-splines*. 1971.

[5] Anh-Quan Cao and Raoul de Charette. "Monoscene: Monocular 3d semantic scene completion". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 3991–4001.

[6] Muqing Cao et al. "DoubleBee: A Hybrid Aerial-Ground Robot with Two Active Wheels". In: *arXiv preprint arXiv:2303.05075* (2023).

[7] Ran Cheng et al. "S3cnet: A sparse semantic scene completion network for lidar point clouds". In: *Conference on Robot Learning*. PMLR. 2021, pp. 2148–2161.

[8] François Chollet. "Xception: Deep learning with depthwise separable convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1251–1258.

[9] Spconv Contributors. *Spconv: Spatially Sparse Convolution Library*. https://github.com/traveller59/spconv. 2022.

[10] Angela Dai, Christian Diller, and Matthias Nießner. "Sg-nn: Sparse generative neural networks for self-supervised scene completion of rgb-d scans". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 849–858.

[11] Dmitri Dolgov et al. "Practical search techniques in path planning for autonomous driving". In: *Ann Arbor* 1001.48105 (2008), pp. 18–80.

[12] Amine Elhafsi et al. "Map-predictive motion planning in unknown environments". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2020, pp. 8552–8558.

[13] David D Fan et al. "Autonomous hybrid ground/aerial mobility in unknown environments". In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2019, pp. 3070–3077.

[14] Martin Garbade et al. “Two stream 3d semantic scene completion”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 2019, pp. 0–0.

[15] Andreas Geiger, Philip Lenz, and Raquel Urtasun. “Are we ready for autonomous driving? the kitti vision benchmark suite”. In: *2012 IEEE conference on computer vision and pattern recognition*. IEEE. 2012, pp. 3354–3361.

[16] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. “3d semantic segmentation with submanifold sparse convolutional networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 9224–9232.

[17] Ruihua Han et al. “NeuPAN: Direct Point Robot Navigation with End-to-End Model-based Learning”. In: *arXiv preprint arXiv:2403.06828* (2024).

[18] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[19] Yuanhui Huang et al. “Tri-perspective view for vision-based 3d semantic occupancy prediction”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023, pp. 9223–9232.

[20] Zilong Huang et al. “Ccnet: Criss-cross attention for semantic segmentation”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 603–612.

[21] Zhuozhu Jian et al. “Putn: A plane-fitting based uneven terrain navigation framework”. In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2022, pp. 7160–7166.

[22] Steven G Johnson et al. *The NLopt nonlinear-optimization package*. 2014.

[23] Sanghun Jung et al. “V-STRONG: Visual Self-Supervised Traversability Learning for Off-road Navigation”. In: *arXiv preprint arXiv:2312.16016* (2023).

[24] Arash Kalantari and Matthew Spenko. “Design and experimental validation of hytaq, a hybrid terrestrial and aerial quadrotor”. In: *2013 IEEE International Conference on Robotics and Automation*. IEEE. 2013, pp. 4445–4450.

[25] Kapil D Katyal et al. “High-speed robot navigation using predicted occupancy maps”. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2021, pp. 5476–5482.

[26] Tarasha Khurana et al. “Point cloud forecasting as a proxy for 4d occupancy forecasting”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 1116–1124.

[27] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[28] Phone Thiha Kyaw et al. “Energy-efficient path planning of reconfigurable robots in complex environments”. In: *IEEE Transactions on Robotics* 38.4 (2022), pp. 2481–2494.

[29] Yiming Li et al. “Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 9087–9098.

[30] Zhiqi Li et al. “Fb-occ: 3d occupancy prediction based on forward-backward view transformation”. In: *arXiv preprint arXiv:2307.01492* (2023).

[31] Junxiao Lin et al. “Skater: A Novel Bi-modal Bi-copter Robot for Adaptive Locomotion in Air and Diverse Terrain”. In: *arXiv preprint arXiv:2403.01991* (2024).

[32] Haisong Liu et al. “Sparsebev: High-performance sparse 3d object detection from multi-camera videos”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 18580–18590.

[33] Yu Liu and Michael S Lew. “Learning relaxed deep supervision for better edge detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 231–240.

[34] Zhijian Liu et al. “Bevfusion: Multi-task multi-sensor fusion with unified bird’s-eye view representation”. In: *2023 IEEE international conference on robotics and automation (ICRA)*. IEEE. 2023, pp. 2774–2781.

[35] Mikhail Martynov et al. “MorphoGear: An UAV with Multi-Limb Morphogenetic Gear for Rough-Terrain Locomotion”. In: *2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE. 2023, pp. 11–16.

[36] Sachin Mehta and Mohammad Rastegari. “Separable self-attention for mobile vision transformers”. In: *arXiv preprint arXiv:2206.02680* (2022).

[37] Xiangyun Meng et al. “Neural Autonomous Navigation with Riemannian Motion Policy”. In: *2019 International Conference on Robotics and Automation (ICRA)*. 2019, pp. 8860–8866. DOI: 10.1109/ICRA.2019.8794223.

[38] Xiangyun Meng et al. “Terrainnet: Visual modeling of complex terrain for high-speed, off-road navigation”. In: *arXiv preprint arXiv:2303.15771* (2023).

[39] Zhenxing Ming et al. “OccFusion: A straightforward and effective multi-sensor fusion framework for 3D occupancy prediction”. In: *arXiv preprint arXiv:2403.01644* (2024).

[40] Neng Pan et al. “Skywalker: A Compact and Agile Air-Ground Omnidirectional Vehicle”. In: *IEEE Robotics and Automation Letters* 8.5 (2023), pp. 2534–2541.

[41] Tong Qin, Peiliang Li, and Shaojie Shen. “Vins-mono: A robust and versatile monocular visual-inertial state estimator”. In: *IEEE Transactions on Robotics* 34.4 (2018), pp. 1004–1020.

[42] Youming Qin et al. “Hybrid aerial-ground locomotion with a single passive wheel”. In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2020, pp. 1371–1376.

[43] Luis Roldao, Raoul de Charette, and Anne Verroust-Blondet. “Lmscnet: Lightweight multiscale 3d semantic completion”. In: *2020 International Conference on 3D Vision (3DV)*. IEEE. 2020, pp. 111–119.

[44] Sahand Sabet et al. “Dynamic modeling, energy analysis, and path planning of spherical robots on uneven terrains”. In: *IEEE Robotics and Automation Letters* 5.4 (2020), pp. 6049–6056.

[45] Amirreza Shaban et al. “Semantic Terrain Classification for Off-Road Autonomous Driving”. In: *Proceedings of the 5th Conference on Robot Learning*. Ed. by Aleksandra Faust, David Hsu, and Gerhard Neumann. Vol. 164. Proceedings of Machine

Learning Research. PMLR, 2022, pp. 619–629. URL: https://proceedings.mlr.press/v164/shaban22a.html.

[46] Eric Sihite et al. "Multi-Modal Mobility Morphobot (M4) with appendage repurposing for locomotion plasticity enhancement". In: *Nature communications* 14.1 (2023), p. 3323.

[47] Shuran Song et al. "Semantic scene completion from a single depth image". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1746–1754.

[48] HJ Terry Suh et al. "Energy-efficient motion planning for multi-modal hybrid locomotion". In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2020, pp. 7027–7033.

[49] Rafal Szczepanski, Tomasz Tarczewski, and Krystian Erwinski. "Energy efficient local path planning algorithm based on predictive artificial potential field". In: *IEEE Access* 10 (2022), pp. 39729–39742.

[50] Qifan Tan et al. "Multimodal dynamics analysis and control for amphibious fly-drive vehicle". In: *IEEE/ASME Transactions on Mechatronics* 26.2 (2021), pp. 621–632.

[51] Du Tran et al. "Learning spatiotemporal features with 3d convolutional networks". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 4489–4497.

[52] Junming Wang et al. "AGRNav: Efficient and Energy-Saving Autonomous Navigation for Air-Ground Robots in Occlusion-Prone Environments". In: *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2024.

[53] Lizi Wang et al. "Learning-based 3D occupancy prediction for autonomous navigation in occluded environments". In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2021, pp. 4509–4516.

[54] Xiaoyu Wang et al. "Path Planning for Air-Ground Robot Considering Modal Switching Point Optimization". In: *2023 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE. 2023, pp. 87–94.

[55] Kasun Weerakoon et al. "Terp: Reliable planning in uneven outdoor environments using deep reinforcement learning". In: *2022 International Conference on Robotics and Automation (ICRA)*. IEEE. 2022, pp. 9447–9453.

[56] Yi Wei et al. "Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 21729–21740.

[57] Tong Wu et al. "Unified Terrestrial/Aerial Motion Planning for HyTAQs via NMPC". In: *IEEE Robotics and Automation Letters* 8.2 (2023), pp. 1085–1092.

[58] Zhaoyang Xia et al. "SCPNet: Semantic Scene Completion on Point Cloud". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 17642–17651.

[59] Zhanteng Xie and Philip Dames. "Drl-vo: Learning to navigate through crowded dynamic scenes using velocity obstacles". In: *IEEE Transactions on Robotics* (2023).

[60] Long Xu et al. “An Efficient Trajectory Planner for Car-Like Robots on Uneven Terrain”. In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2023, pp. 2853–2860. DOI: 10.1109/IROS55552.2023.10341558.

[61] Shuangjie Xu et al. “Sparse cross-scale attention network for efficient lidar panoptic segmentation”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 3. 2022, pp. 2920–2928.

[62] Zhefan Xu et al. “A real-time dynamic obstacle tracking and mapping system for UAV navigation and collision avoidance with an RGB-D camera”. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2023, pp. 10645–10651.

[63] Zhefan Xu et al. “Onboard dynamic-object detection and tracking for autonomous robot navigation with RGB-D camera”. In: *IEEE Robotics and Automation Letters* 9.1 (2023), pp. 651–658.

[64] Xu Yan et al. “Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 4. 2021, pp. 3101–3109.

[65] Chenyu Yang et al. “BEVFormer v2: Adapting Modern Image Backbones to Bird’s-Eye-View Recognition via Perspective Supervision”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 17830–17839.

[66] Jiawei Yao et al. “Ndc-scene: Boost monocular 3d semantic scene completion in normalized device coordinates space”. In: *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE Computer Society. 2023, pp. 9421–9431.

[67] Fisher Yu and Vladlen Koltun. “Multi-scale context aggregation by dilated convolutions”. In: *arXiv preprint arXiv:1511.07122* (2015).

[68] Ruibin Zhang et al. “Autonomous and adaptive navigation for terrestrial-aerial bimodal vehicles”. In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 3008–3015.

[69] Ruibin Zhang et al. “Model-Based Planning and Control for Terrestrial-Aerial Bimodal Vehicles with Passive Wheels”. In: *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2023, pp. 1070–1077. DOI: 10.1109/IROS55552.2023.10342188.

[70] Xinyu Zhang et al. “A Multi-modal Deformable Land-air Robot for Complex Environments”. In: *arXiv preprint arXiv:2210.16875* (2022).

[71] Xinyu Zhang et al. “Coupled Modeling and Fusion Control for a Multi-modal Deformable Land-air Robot”. In: *arXiv preprint arXiv:2211.04185* (2022).

[72] Zhilu Zhang and Mert Sabuncu. “Generalized cross entropy loss for training deep neural networks with noisy labels”. In: *Advances in neural information processing systems* 31 (2018).

[73] Boyu Zhou et al. “Raptor: Robust and perception-aware trajectory replanning for quadrotor fast flight”. In: *IEEE Transactions on Robotics* 37.6 (2021), pp. 1992–2009.

[74] Boyu Zhou et al. “Robust and efficient quadrotor trajectory generation for fast autonomous flight”. In: *IEEE Robotics and Automation Letters* 4.4 (2019), pp. 3529–3536.

[75] Xin Zhou et al. “Ego-planner: An esdf-free gradient-based local planner for quadrotors”. In: *IEEE Robotics and Automation Letters* 6.2 (2020), pp. 478–485.

[76] Xin Zhou et al. “Ego-swarm: A fully autonomous and decentralized quadrotor swarm system in cluttered environments”. In: *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE. 2021, pp. 4101–4107.

[77] Xin Zhou et al. “Swarm of micro flying robots in the wild”. In: *Science Robotics* 7.66 (2022), eabm5954.

[78] Sicheng Zuo et al. “Pointocc: Cylindrical tri-perspective view for point-based 3d semantic occupancy prediction”. In: *arXiv preprint arXiv:2308.16896* (2023).