Université du Québec
à Chicoutimi

ENERGY-AWARE INFORMATIVE PATH PLANNING AND EMERGENCY SAFE LANDING APPROACH FOR UAVS IN SEARCH AND RESCUE MISSIONS

BY AHMAD MEREI

THESIS PRESENTED TO L'UNIVERSITÉ DU QUÉBEC À CHICOUTIMI IN PARTIAL FULFILLMENT OF THE REQUIERMENTS FOR THE DEGREE OF PHILOSOPHIÆ DOCTOR (PH.D.) IN THE SUBJECT OF INFORMATIQUE

QUÉBEC, CANADA

ABSTRACT

Uncrewed Aerial Vehicles (UAVs) are increasingly deployed in healthcare, disaster response, and search-and-rescue (SAR) operations, where rapid decisions in complex and uncertain environments can save lives. However, their effectiveness is constrained by three major challenges.

The first is efficiently exploring an Area of Interest (AOI) to identify and prioritize the most informative Regions of Interest (ROI) in missions that are both time-critical, where every minute can determine survival outcomes, and energy-critical, where limited battery capacity restricts flight endurance. The second is coordinating multiple UAVs so that tasks are distributed in a balanced way across ROIs while avoiding redundant exploration and wasted energy. The third is guaranteeing safe autonomous landings, whether in cluttered areas near survivors to support communication with first responders, or in unpredictable situations such as GPS-denied environments and sudden emergencies where immediate landing is required.

Single-UAV and multi-UAV informative path planning (IPP and MIPP) approaches have advanced, but they still face important limitations in practice: effectively balancing energy and time budgets while prioritizing information gain, and coordinating multiple vehicles to avoid redundant effort and uneven workload. Likewise, existing landing systems often fail to ensure obstacle avoidance and survivor-aware decision-making under emergency conditions, and many rely on pre-defined landing pads that are not available in real SAR scenarios.

This thesis addresses these gaps with three contributions. First, it introduces an *Energy-Aware Informative Path Planner with two-step lookahead (EA-IPP2n)*, designed to maximize information gain while explicitly accounting for both time and energy as mission-critical resources. Second, it extends this framework to cooperative operations through a *V-shape Area Partitioning (VAP)* strategy and an *Energy-Aware Multi-UAV Informative Path Planner (EA-MIPP)*, distributing ROI tasks across agents in a workload-balanced manner while handling static obstacles and Non-Flying Zones (NFZ). Third, it develops an *Emergency Safe Landing System (ESLS)* that integrates mapping, localization, and object detection to autonomously identify safe landing zones, including an *ESLZD-SAR* mode that prioritizes landings near detected survivors.

Together, these contributions advance a unified framework for UAV autonomy that is both resource-aware and resilient in mission execution. The proposed methods improve information collection in AOIs, enable scalable multi-UAV collaboration across ROIs, and support safe emergency landings. By combining energy-aware path planning with adaptive landing strategies, this work enhances the operational capabilities of UAVs in healthcare and SAR missions.

RÉSUMÉ

Les véhicules aériens sans pilote (UAV – Uncrewed Aerial Vehicles) sont de plus en plus utilisés dans les domaines de la santé, de la gestion des catastrophes et des opérations de recherche et sauvetage (SAR). Dans ces environnements, la prise de décision rapide est essentielle. Elle s'effectue dans des environements complexes et incertains, ou chaque instant compte et peut sauver des vies. Cependant, leur efficacité demeure limitée par trois défis majeurs.

Le premier défi consiste à explorer efficacement une zone d'intérêt (AOI – Area of Interest). L'objectif est d'identifier et de prioriser les régions informatives importantes (ROI – Regions of Interest). Ces missions sont à la fois critiques en temps, où chaque minute peut déterminer les chances de survie, et critiques en énergie, où la capacité limitée des batteries restreint l'endurance de vol. Le deuxième défi concerne la coordination de plusieurs UAVs. Il s'agit de répartir les tâches de manière équilibrée entre les différentes ROI, tout en évitant l'exploration redondante et le gaspillage d'énergie. Enfin, le troisième défi réside dans la garantie d'atterrissages autonomes en toute sécurité. Il s'agit, d'une part, la capacité à se poser dans des zones encombrées à proximité de survivants, afin de faciliter la communication avec les premiers intervenants. D'autre part, le système doit gérer des situations imprévisibles telles que les environnements sans GPS ou des urgences nécessitant un atterrissage immédiat.

Les approches existantes de planification de trajectoire informative, qu'elles soient mono-UAV ou multi-UAV, ont permis des avancées notables. Cependant elles peinent encore à surmonter certaines limites en pratiques: l'utilisation optimale de l'énergie, la maximisation de la couverture de mission, ainsi que la coordination efficace de plusieurs véhicules. De même, les systèmes d'atterrissage existants échouent souvent à éviter les obstacles. Par ailleurs, l'intégration des informations relatives aux survivants dans les algorithmes de décision en situation d'urgence reste partielle. De plus, ces systèmes s'appuient fréquemment sur des zones d'atterrissage pré-définies, qui sont rarement disponibles dans les scénarios réels de recherche et sauvetage (SAR).

Cette thèse répond à ces défis à travers trois contributions principales. Premièrement, elle introduit un planificateur de trajectoire informative à deux étapes anticipées et à conscience énergétique (noté EA-IPP2n). Ce planificateur est conçu pour maximiser le gain d'information tout en considérant explicitement le temps et l'énergie comme ressources critiques de mission. Deuxièmement, elle étend ce cadre aux opérations coopératives. Pour cela, elle propose une stratégie de partitionnement en forme de V (noté VAP) et un planificateur de trajectoire informative multi-UAV à conscience énergétique (noté EA-MIPP). Cette approche permet une répartition équilibrée des tâches entre les ROI, tout en gérant les obstacles statiques et les zones de non-vol. Troisièmement, elle développe un système d'atterrissage d'urgence sécurisé (noté ESLS). Ce systeme intégre plusieurs modules: la cartographie, la localisation et la détection d'objets afin d'identifier de manière autonome des zones d'atterrissage sûres. De

plus, un mode spécialisé est prevu pour privilégier les atterrissages à proximité des survivants détectés.

Ensemble, ces contributions forment un cadre unifié pour l'autonomie des UAVs, à la fois conscient des ressources et résilient dans l'exécution des missions. Les méthodes proposées améliorent la collecte d'informations dans les AOI, permettent une collaboration multi-UAV et assurent des atterrissages d'urgence sécurisés. En combinant planification de trajectoire énergétique et stratégies d'atterrissage adaptatives, cette recherche renforce les capacités opérationnelles des UAV dans les missions de santé et de recherche et sauvetage.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ALGORITHMS

LIST OF ABBREVIATIONS

ACO	Ant Colony Optimization
AI	Artificial Intelligence
AOI	Area of Interest
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CPP	Coverage Path Planning
DNN	Deep Neural Network
DQN	Deep Q-Network
DRL	Deep Reinforcement Learning
DWA	Dynamic Window Approach
EA-IPP2n	Energy-Aware Informative Path Planning (Two-Hop Lookahead)
EA-MIPP	Energy-Aware Multi-UAV Informative Path Planning
EKF	Extended Kalman Filter
ESLS	Emergency Safe Landing System
ESLZD	Emergency Safe Landing Zone Detection
ESLZD-SAR	Emergency Safe Landing Zone Detection for Search and Rescue
FOV	Field of View
FPV	First Person View (camera)
GA	Genetic Algorithm
GCS	Ground Control Station
GPU	Graphics Processing Unit
GPS	Global Positioning System
HIPP	Heuristic Informative Path Planning

HSV	Hue–Saturation–Value (color model)
IBVS	Image-Based Visual Servoing
IMU	Inertial Measurement Unit
IPP	Informative Path Planning
KPI	Key Performance Indicator
LiDAR	Light Detection and Ranging
Li-Po	Lithium-Polymer (battery)
LoS	Line of Sight
MAV	Micro Aerial Vehicle
MAVLink	Micro Air Vehicle Link
MAVROS	MAVLink ROS Package
MCTS	Monte Carlo Tree Search
MDP	Markov Decision Process
ML	Machine Learning
MIPP	Multi-UAV Informative Path Planning
NFZ	Non-Flying Zone
ODA	Obstacle Detection and Avoidance
OP	Orienteering Problem
PRM	Probabilistic Roadmap
PT	Processing Time
QoC	Quality of Coverage
RGB	Red–Green–Blue (image)
RGB-D	Red–Green–Blue–Depth (camera)
ROI	Region of Interest
ROS	Robot Operating System

RTAB-Map	Real-Time Appearance-Based Mapping
RRT	Rapidly-Exploring Random Tree
RRT*	Optimal Rapidly-Exploring Random Tree
SITL	Software-in-the-Loop
SLAM	Simultaneous Localization and Mapping
SLZ	Safe Landing Zone
SAR	Search and Rescue
SC1	Scenario 1 (used in performance evaluation)
SC2	Scenario 2 (used in performance evaluation)
UAV	Uncrewed Aerial Vehicle
UGV	Uncrewed Ground Vehicle
USV	Uncrewed Surface Vehicle
VAP	V-shape Area Partitioning
VIO	Visual-Inertial Odometry
YOLO	You Only Look Once (object detector)

LIST OF SYMBOLS

(u,v)	Pixel coordinates in YOLOv5 frame
(x^*,y^*)	Center of selected safe landing zone
(x_N,y_N)	Next UAV position above the landing zone
α,β	Weights in cost evaluation function
$\bar{r}$	Normalized average reward of a candidate transition
ΔE	Incremental energy consumed between two nodes
ΔT	Incremental time between two nodes
$\Delta x,\Delta y$	Real-world displacement from UAV to landing zone
$\Delta\theta$	Heading change or orientation difference
δ^*	Selected optimal displacement direction
δ_i	Candidate displacement direction for avoidance
η	Weighting coefficient in the cost function
γ	Energy cost per degree turned
λ	Energy cost per meter traveled
$\mathbf{x},\mathbf{y}$	Node coordinate vectors (X and Y positions)
$\mathscr{B}$	Set of bounding boxes detected by YOLOv5
$\mathscr{G}=(V,E)$	Graph representation of the environment
$\mathscr{N}(n)$	Neighborhood of node n
$\mathscr{P}$	Set of candidate paths in the planning algorithm
Ω	Global area of interest domain
ω	UAV rotational rate
$\Omega_j(\Phi)$	Partitioned ROI assigned to UAV u_j
Ω_{ROI}	Set of pixels/points belonging to detected ROI

$\Phi = \{\phi_1, \ldots, \phi_{m-1}\}$	Partition cut angles for VAP
ϕ_h, ϕ_v	Horizontal and vertical field of view (camera)
ρ_{safe}	Minimum safe distance to obstacles
σ	Standard deviation (used in performance results)
$\theta(x,y)$	Polar angle of point (x,y) relative to starting point
θ_i	UAV heading angle at node i
$\vec{p}_t$	UAV position at time t
$\vec{v}_t$	UAV velocity at time t
$\vec{w}_{t+1}$	Next waypoint
$A(ROI)_{\text{tot}}$	Total ROI area
A_j	ROI area of partition j
B	Set of bounding boxes detected by YOLO11
C_{kj}	Cost of transition (k, j) in EA-MIPP
$D(x,y)$	Distance transform (distance to nearest obstacle in ESLZD)
d_{ij}	Distance between nodes i and j
E_i	Energy at step or node i
E_{bud}	Energy budget constraint
E_{cons}	Actual energy consumed during mission
E_{norm}	Normalized energy consumption
E_{total}	Total stored UAV battery energy
f	Camera focal length
h	UAV flight altitude
J_{area}	Objective function for area balance across partitions
L_p	Total path length
$M(x,y)$	Binary mask of ROI ($M = 1$ for ROI, $M = 0$ otherwise)

n_0	Starting position of UAVs
N_T, N_E	Normalized time and energy terms
N_v	Number of visited nodes
$O(x,y)$	Binary occupancy map in ESLZD (1 = free, 0 = obstacle)
p, p^*	Candidate path and optimal path
P_t	Processing time (planner computation)
R	Cumulative collected reward
$r(n)$	Reward of node n
R_i	Reward at node or region i
R_{avg}	Average collected reward per visited node
R_{norm}	Normalized total reward
r_{safe}	Safety threshold radius ($r_{\text{safe}} \geq 3r_{\text{UAV}}$)
R_{tot}	Total collected reward
r_{UAV}	Physical UAV radius
T_i	Time at step or node i
T_{bud}	Time budget constraint
T_{comp}	Completion time of UAV mission
T_{norm}	Normalized time consumption
$U = \{u_1, \ldots, u_m\}$	UAV team of m agents
v	UAV forward speed
W, H	Real-world width and height of the image footprint at altitude h

DEDICATION

To my beloved parents,

To my dear sister and brothers,

To my dear nieces and nephews,

To my cherished friends,

Your endless love, unwavering support, and constant encouragement have been my greatest strengths. Thank you from the bottom of my heart.

ACKNOWLEDGEMENTS

I first and foremost thank God Almighty, the creator of knowledge, for granting me the perseverance and determination to complete this doctoral journey.

I am sincerely grateful to my supervisors, Prof. Hamid Mcheick and Dr. Alia Ghaddar, for their guidance and constructive feedback during my studies. Their advice helped shape this work while giving me the freedom to explore my own ideas and directions.

I also wish to thank Université du Québec à Chicoutimi for providing the academic environment in which this research was carried out. A special note of appreciation goes to the MIST Laboratory at Polytechnique Montréal, where I completed an internship under the supervision of Prof. Giovanni Beltrame. Working on the Emergency Safe Landing System (ESLS) with his team was a rewarding experience that enriched my skills and broadened my perspective.

I am thankful to my colleagues, lab mates, and friends for their encouragement, discussions, and collaboration, which lightened the challenges of this journey. Their support made the demanding moments more manageable and the achievements more meaningful.

Finally, I owe my deepest gratitude to my family. To my parents, for their endless prayers, sacrifices, and unwavering belief in me, and to my siblings, for their constant encouragement and support. This accomplishment would not have been possible without their love and patience throughout these years.

CHAPTER I

INTRODUCTION

1.1 OVERVIEW

Uncrewed Aerial Vehicles (UAVs), commonly known as drones, have become increasingly integrated into diverse domains due to their autonomy, maneuverability, and ability to access hazardous or hard-to-reach environments. A key component enabling their operational efficiency is advanced path planning, which determines optimal routes while considering constraints such as energy, time, and safety. They are used in agriculture, environmental monitoring, search and rescue (SAR) operations, infrastructure inspection, and planetary exploration [1, 2, 3, 4]. These application areas demonstrate the potential of UAVs as rapid, flexible, and cost-effective tools. At the same time, they highlight a recurring challenge: autonomous mission execution must remain efficient and effective under strict energy, time, and safety constraints.

Among the various path planning problems, *Informative Path Planning* (IPP) has attracted significant attention. IPP methods explicitly balance travel cost against expected information gain: the UAV is guided to positions that maximize utility under resource limitations. In formal terms, IPP is often framed as a variant of the Orienteering Problem (OP), in which an agent selects a subset of nodes to visit within bounded budgets (e.g., time) to maximize cumulative reward [5, 6, 7]. In UAV applications, IPP has been studied for mapping, exploration, and surveillance, where information efficiency is as critical as coverage [8, 1, 2].

Scaling from a single UAV to multi-UAV deployments offers the possibility of wider-area coverage, faster mission completion, and more robust data collection. Such missions are especially relevant for SAR, disaster response, and environmental monitoring [9, 10, 11]. However, multi-UAV coordination introduces its own challenges: risk of collisions, redundant coverage, high communication overhead, and imbalanced workloads. Without explicit strategies—such as area partitioning and workload-aware allocation—efficiency and safety are difficult to guarantee. Furthermore, real deployments must comply with Non-Flying Zones (NFZs) and contend with static obstacles (e.g., buildings), adding further complexity to mission planning.

Autonomy also depends on robust mechanisms for handling emergencies. Failures in GPS, communication, or onboard sensors may force unplanned descents. In cluttered or GPS-denied environments, emergency landing becomes especially challenging: existing approaches often assume controlled conditions, such as predefined landing pads or fiducial markers, which limits their real-world applicability [12, 13, 14]. Developing vision-based methods that can dynamically identify safe landing zones remains an open and pressing issue for UAV autonomy.

1.2 RESEARCH PROBLEM

The overarching problem addressed in this thesis can be stated as follows:

> *How can UAVs autonomously plan efficient, safe, and energy-aware paths for single UAV or multi-UAVs, while guaranteeing safe landing under emergency conditions or in SAR missions?*

This question highlights three tightly coupled dimensions of UAV autonomy: efficient single-UAV path planning under dual energy and time budgets, coordination strategies for multi-UAV deployments, and autonomous emergency landing in unstructured environments.

1.3 HYPOTHESES

From this research problem, the following general hypothesis and sub-hypotheses are formulated. Each hypothesis is testable through quantitative evaluation and directly corresponds to the experiments presented in later chapters.

General Hypothesis: Autonomous search and rescue missions using UAVs can be reliably achieved by a unified pipeline that integrates informative exploration, cooperative coverage, and emergency safe landing while operating under strict time and energy constraints.

- **Sub-hypothesis 1 (Single-UAV informative path planning):** An energy-aware informative path planning algorithm that accounts for both time and energy budgets, addressing dual-orienteering problem, enables a single UAV to improve path efficiency, collecting higher reward while adhering to budgets, compared to baseline planning methods (Chapter 4).

- **Sub-hypothesis 2 (Multi-UAV informative path planning):** Partitioning the area of interest into balanced V-shape sectors combined with energy-aware informative planning enables multi-UAV teams to achieve balanced coverage, avoid path overlap, obstacles and non flying zones (Chapter 5).

- **Sub-hypothesis 3 (Emergency safe landing):** A vision-based safe landing zone detection algorithm combined with point cloud–based obstacle-aware descent planning, enables UAVs to autonomously perform safe emergency landings in GPS-denied and cluttered environments in general situations and during search and rescue missions (Chapter 6).

1.4 THESIS CONTRIBUTIONS

This thesis contributes three complementary components that together form a resource-aware UAV mission pipeline (Figure 1.1):

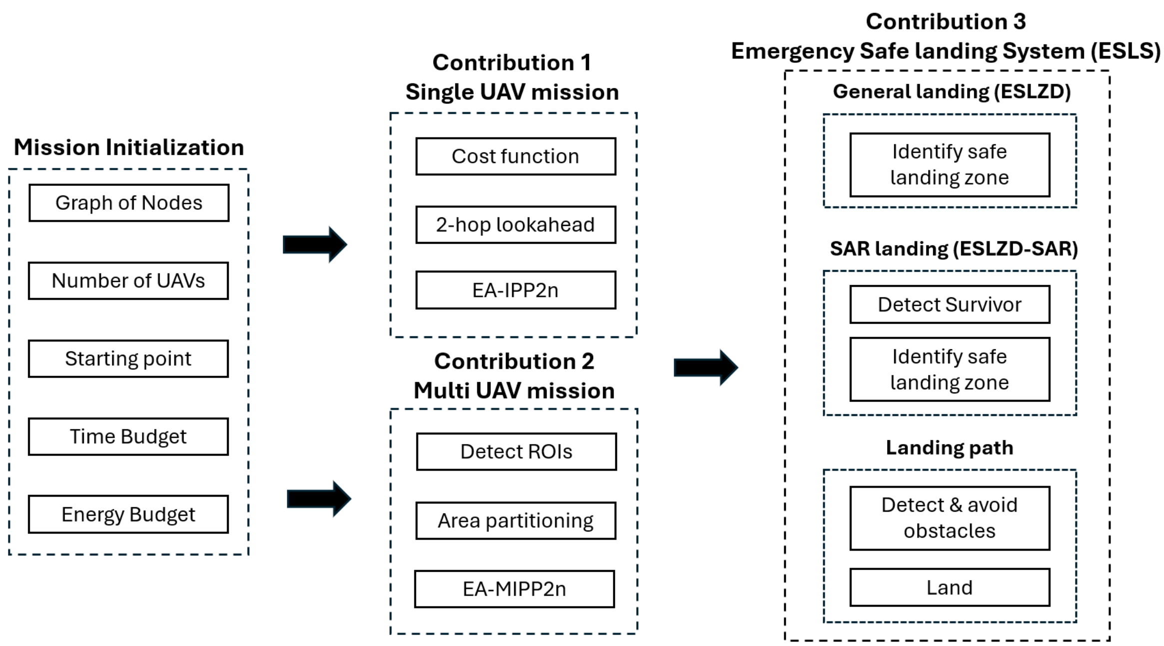


Figure 1.1 Resource-aware UAV mission pipeline.

- **Energy-Aware Informative Path Planer(EA-IPP2n):** A single-UAV planner that accounts for both distance and turning costs under dual time and energy budgets. It employs a two-hop look-ahead decision mechanism with normalized cost–reward tradeoffs and is evaluated against multiple baselines (Chapter 4).

- **V-shape Area Partitioning for Multi-UAV IPP (VAP-EAMIPP):** A cooperative framework extending EA-IPP2n to multi-UAV missions. It partitions the area into balanced V-shaped angular sectors to minimize redundancy and integrates NFZ and obstacle handling for efficient planning (Chapter 5).

- **Emergency Safe Landing System (ESLS):** A vision-based landing pipeline that integrates RTAB-MAP SLAM with visual–inertial odometry for real-time localization and mapping. It employs YOLO in conjunction with binary occupancy mapping to detect safe landing zones and utilizes point cloud data for 3D obstacle avoidance during descent. The system functions autonomously without GPS or external infrastructure and has been validated in realistic Gazebo/ArduPilot simulations. It supports two operational modes: *ESLZD* (obstacle-free landing) and *ESLZD-SAR* (landing in proximity to detected survivors) (Chapter 6).

1.5 THESIS ORGANIZATION

The remainder of this thesis is organized as follows:

- **Chapter 2** introduces UAV technologies, their types and applications. Overviews the UAV path planning problem, highlights the associated challenges across different operational domains, and presents the key performance metrics commonly used to evaluate UAV missions.
- **Chapter 3** reviews the state of the art in UAV applications for healthcare missions. This chapter surveys relevant path planning algorithms tailored to such contexts and presents a detailed overview of obstacle detection and avoidance strategies reported in the literature.
- **Chapter 4** presents the EA-IPP2n algorithm and evaluates its performance in single-UAV scenarios and compares it with existing baselines.
- **Chapter 5** extends the framework to multi-UAV missions through the VAP-EAMIPP approach and compares it with existing baseline.

- **Chapter 6** Proposes the Emergency Safe Landing System (ESLS), its two operational modes, and evaluation results.
- **Chapter 7** presents system-level integration of the proposed methods and reports SAR mission simulation demonstrating their combined effectiveness.
- **Chapter 8** concludes the thesis by summarizing contributions and outlining directions for future research.
- **Chapter 9** lists the publications resulting from this PhD thesis.

CHAPTER II

BACKGROUND

2.1 UNCREWED AERIAL VEHICLES (UAVS)

2.1.1 OVERVIEW

In recent years, UAVs have seen a rapid increase in use across various tasks. These aircraft operate without a pilot onboard and are controlled either remotely or by onboard computers. As a result, this technology has opened up countless opportunities across numerous fields.

UAVs are available in different shapes and sizes, with varying types designed to cater to different purposes. Some are small enough to fit in the palm of your hand, while others are large and can carry heavy payloads. There are fixed-wing UAVs that resemble traditional airplanes, rotary wing UAVs that operate like helicopters, and even hybrid UAVs that combine both designs. These differences in UAV types and configurations make them versatile tools that can be used for a wide range of applications.

2.1.2 UAV TYPES

There are three main types of UAVs, which are Rotary Wing, Fixed Wing, and Hybrid. Each type has different features, sizes, and specifications such as footprint, weight, and power support (gas, electric, nitro, solar, etc.).

Rotary Wing UAVs, also known as rotary-wings UAVs, allow for vertical take-off and landing and can hover over a fixed location for continuous cellular coverage. This type of aerial platform can be classified into single-rotor (helicopter) and multi-rotor (quadcopter and hexacopter). Quadcopters are affordable and have good maneuverability. They're widely used in surveillance missions, such as in confined spaces, close-range inspection tasks, marine settings, terrestrial areas with steep terrain, or extensive vegetation cover.

Fixed Wing UAVs are heavier and require more battery power. However, they have the advantage of carrying higher payloads. They are ideal for use in large areas due to their high durability and long flight time duration. However, they cannot hover or focus on the same scene for a long period of time due to their high speed. They cannot stop or slow down the speed during the mission. Thus, fixed-wing UAVs cannot be used as remote flying camera controllers as they need high-speed sensors to survey.

Hybrid UAVs can switch between two modes of operation: in one mode, they can float like helicopters, and in the other mode, they can fly like fixed-wing UAVs.

The three main types of UAVs: rotary wing (Figure 2.1a), fixed wing (Figure 2.1b), and hybrid (Figure 2.1c) [15].

(a) Rotary Wing (b) Fixed Wing (c) Hybrid

Figure 2.1 UAV types.

2.1.3 APPLICATION DOMAINS AND MISSION CHALLENGES

This technology has been implemented in many applications, including healthcare [3, 16, 17, 18], smart agriculture [19], photogrammetry [20], monitoring [21], rescue and search [22], and other domains. The use of UAVs in the healthcare field has emerged as one of the most interesting topics in recent years [23]. UAVs have been used for medical supplies delivery [24], organ transfer [25], emergency and first aid delivery [26], and monitoring patients [27]. In addition, UAVs are gaining widespread attention, especially during the pandemic [28, 29].

UAVs have become an important means of providing transportation, monitoring, and delivery of supplies such as PCR tests and kits in remote areas [30]. A single UAV or a fleet of UAVs can deliver medical supplies to rural areas that are inaccessible by road or air transport. UAVs are also noted for their low cost and simplicity [31]. UAVs can bypass difficult obstacles such as buildings and other structures. Furthermore, UAVs can help automate work and reduce human interaction so that the tasks can be fully automated in some cases.

In any application domain, a path must be planned for the UAV to achieve the mission goal. Different patterns and techniques are used to plan a path to maximize coverage in the environment (known as coverage path planning methods) or maximize the utility of the data collection (known as informative path planning methods). Each path planning method could be used for a specific type of mission and faces different challenges such as wind, obstacles, UAV battery lifetime, collision avoidance, or birds. The efficiency of a planned path can be evaluated using different performance metrics, such as energy consumption and completion time.

2.2 PATH PLANNING FOR UAVS

2.2.1 OVERVIEW

UAVs have the potential to revolutionize healthcare by enabling the delivery of medical supplies and equipment to remote or hard-to-reach areas and in some cases acting as a lifesaver. They can also be used for telemedicine, allowing doctors to remotely diagnose and treat patients using cameras and other equipment mounted on the UAV.

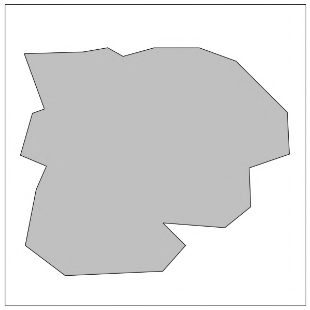
Even more UAVs are capable of searching for patients in unknown areas and under harsh circumstances using single or swarm of UAVs. Path planning for UAVs in healthcare missions is an important area of research, as it involves determining the optimal path for the UAV to reach its destination while avoiding obstacles and ensuring the safety of the aircraft and the parcel [32, 33].

Coverage Path Planning (CPP) and Informative Path Planning (IPP) are powerful techniques that can be applied in various fields, including healthcare missions. These techniques allow for the efficient and effective use of UAVs to cover an area of interest and gather valuable data, track patients, and locate individuals who require assistance.

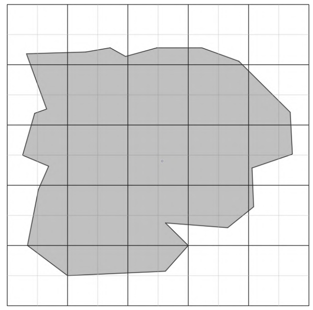
In healthcare missions, UAVs can be utilized to cover large areas such as hospitals, nursing homes, and remote communities to gather data, or like in COVID-19 pandemic where UAVs cover areas to search for people who are not wearing masks. By using CPP and IPP, UAVs can navigate the area in a safe and efficient manner while considering the dynamic constraints of the healthcare mission, such as obstacles, patients' privacy, and safety.

UAV path could be planned before the mission (offline), during the mission (online), or both could be combined [34]. The area in most cases, is divided into a grid (Figure 2.2b) and each cell is fitted to the UAV footprint (Figure 2.2c). To plan an offline path, planners scan the area from a top view, as pictured by Figure 2.2a.

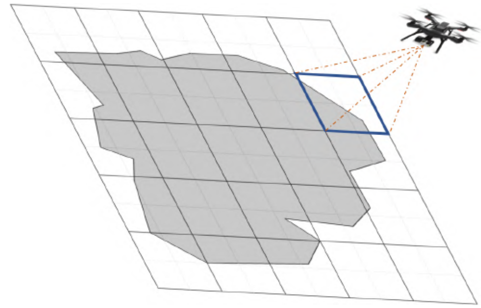
For online path planning, UAV vision and flight control require specific equipment to offer real-time visual data in order to generate updated mapping and to overcome mission challenges. Challenges addressed in recent works include power consumption, mission cost,



(a) AOI Top View



(b) Grid Map



(c) UAV Footprint

Figure 2.2 AOI top view and grid division.

time, path length, number of turns, wind, and quality of coverage. These challenges vary according to the mission type, and path status.

2.2.2 COVERAGE PATH PLANNING

CPP is a problem in route planning that aims to determine the best path that covers all points of interest (AoI) in an environment [35] (see Figure 2.3). Several research studies have investigated CPP and have presented a variety of path planning solutions, considering different parameters such as the geometry of the area, whether the plan is online or offline, the number of UAVs, completion time, and energy consumption.

Various methods have been proposed for calculating a low-cost coverage path, including the mirror mapping method [36], context-aware UAV mobility [37], and decomposition-based methods such as classical precise cellular decomposition [38], Boustrophedon decomposition [39], landmark-based topological coverage [40], and grid-based approaches using trapezoidal patterns [41, 42]. These methods all involve breaking down the area into smaller cells, which are then visited in an efficient way to fill the free space.

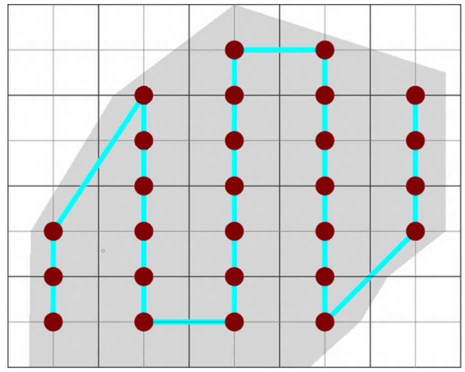


Figure 2.3 Coverage path planned for a UAV.

Motion patterns such as back-and-forth [43], spiral or circular motions [44] are used to cover the area efficiently while considering factors such as energy consumption, time and distance, camera resolution and field of view to minimize overlap and recurrences. Some studies have used multiple UAVs [45, 46] or mathematical probabilistic principles through integer linear programming [47] to optimize results and reduce completion time.

2.2.3 INFORMATIVE PATH PLANNING

IPP is a technique that uses data and algorithms to optimize the flight paths of UAVs for specific tasks such as in healthcare missions. This can involve using information about the environment, such as temperature and heartbeat data, to plan paths and locate patients. By minimizing the time and energy required to complete a task, IPP can lead to a number of benefits in healthcare missions. For example, it can enable UAVs to deliver defibrillators more quickly to remote or hard-to-reach locations, potentially saving lives. Additionally, IPP can improve the safety and reliability of search and rescue operations by helping UAVs quickly identify the location of survivors and provide critical information to first responders.

The IPP technique allows UAVs to be aware of the environment and prioritize important spots, making them more efficient in their missions. However, key challenges include the strong coupling of multiple layers of decisions such as location selection, sensor platform allocation and information processing along the path.

Figure 2.4 presents a sample of an informative path that has been specifically designed for the UAV, with the use of color coding to indicate the level of importance assigned to each location. As can be observed, locations denoted in pink and red are considered to be of higher significance than the other nodes.

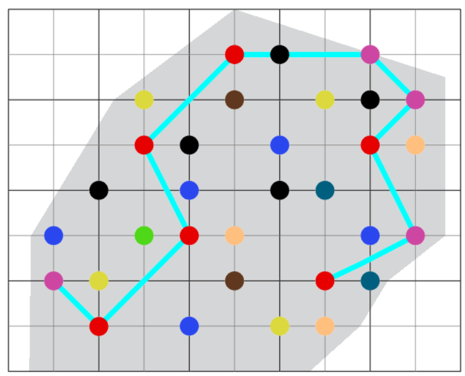


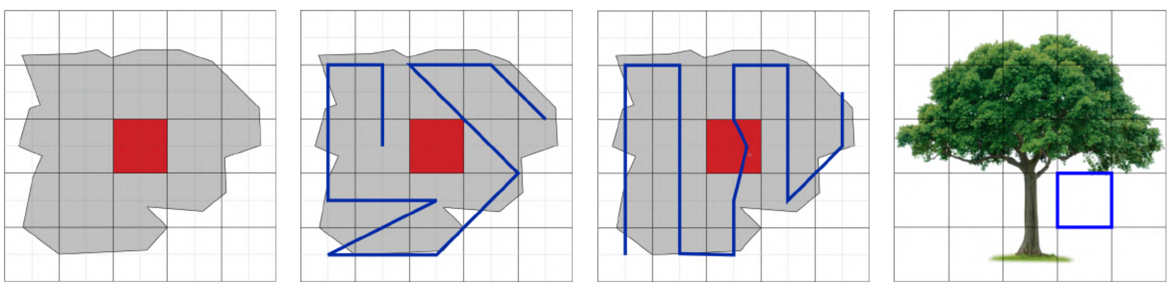
Figure 2.4 Informative path planned for a UAV.

Examples of IPP methods include the algorithm proposed in [48], which plans an informative path for multiple UAVs and takes into account wind, using an occupancy grid to model weed presence on farmland [49], using global viewpoint selection and evolutionary optimization to refine the UAV path while satisfying dynamic constraints [49], mapping strategies for UAV-based agricultural monitoring [50], an IPP framework that maps either discrete or continuous targets on the terrain using a probabilistic altitude-dependent sensor [8], an IPP method that uses terrain maps to create paths in continuous 3D space during a mission using a combination of the recursive Bayesian mapping method and an optimization strategy with informed initialization[8], and an IPP method based on novel criteria that guide the UAV to its optimal next position, chosen using previously collected information in a mathematically elegant and computationally tractable way [51].

Accordingly, the use of CPP and IPP in healthcare missions can have a significant impact on the delivery of medical care, as it enables the efficient and effective use of UAVs to cover large areas and gather valuable data, track patients, and locate individuals who require assistance. This can help reduce response times, improve the quality of care, and save lives.

2.3 PATH PLANNING & COLLISION AVOIDANCE PROBLEM

During the mission, an obstacle may block the UAV's path and prevent it from completing its flight. It may also damage the UAV while it is hovering in a tracking mission. The obstacle could be static or dynamic. Static obstacles are like trees and signboards. They can be located in the area grid map from the top view (see the red cell in Figure 2.5a).



(a) Static obstacle marked by a red cell

(b) Omit obstacle cell

(c) Passing through red cell

(d) UAV position after scanning the frontal view

Figure 2.5 Static obstacle detection and avoidance examples.

The grid cell where the obstacle is located could be avoided either partially or completely by the UAV. In some works [52], the whole cell is completely omitted during the mission, as shown in Figure 2.5b. However, in other works [53], frontal obstacles are avoided by scanning the frontal view before reaching the red cell (Figure 2.5c), in order for the UAV to decide about its new position (Figure 2.5d). The obstacle could be detected from different perspectives and angles using different sensors outlined in the following sections.

Dynamic obstacles are flying birds, another UAV, or any other moving object that could intercept the UAV path [54, 55]. They could be avoided in a 2D or 3D environment, but the 3D ones are more realistic. Dealing with dynamic obstacles is typically more challenging than static ones due to several factors, such as the obstacle's speed and motion. A UAV may also be intercepted by one or more obstacles from different directions.

To ensure safety, UAVs are equipped with sensors to monitor the environment, such as laser scanners, sonar, radar, lidar, and cameras (stereo, monocular, FPV, etc.) [56]. One important aspect of this monitoring process is to avoid collisions [57, 58]. The sensor can accomplish many tasks by generating 2D or 3D maps that help in detecting the obstacle's characteristics, for instance, Lane boundary shape, vehicles, people, distance estimation from the UAV to obstacles, and so on.

2.4 EVALUATION METRICS

Key performance indicators (KPIs) are essential for evaluating the effectiveness of a UAV path-planning approach. These indicators include path length, number of turns, turning angles, rotational rate, mission completion time, UAV energy consumption, mission cost, and coverage quality. Several models for energy consumption and mission completion time have been proposed in the literature [59, 44, 41, 53, 42] include some models for energy consumption and mission completion time. Additionally, alternative models for completion time are available in [60, 61], providing further evaluation methods for route efficiency.

In a grid-based representation, the environment is modeled as a graph $G(V,E)$, where V denotes the set of vertices and E represents the set of edges. Each vertex corresponds to the center of a grid cell and is defined by its coordinates in a Cartesian plane, as illustrated in Figure 2.6.

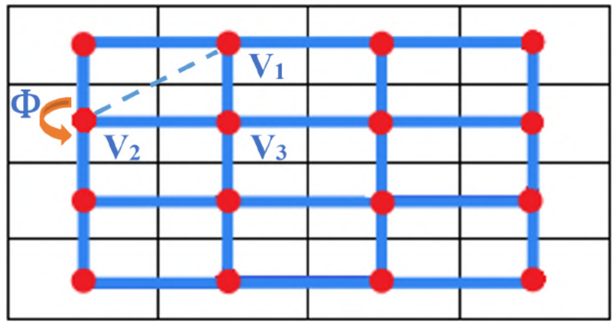


Figure 2.6 Graph representation: red points are vertices (nodes), blue lines are edges (connections).

The energy consumption of a UAV is influenced by the rate at which it turns [53, 62]. Minimizing the number of turns throughout the mission is therefore an important consideration. If a UAV moves from vertex v_1 to v_2 and turns before proceeding to v_3 (see Figure 2.6), the turning angle can be calculated as follows:

$$\Phi_{123} = \pi - \cos^{-1}(\angle v_1 v_2 v_3) \tag{2.1}$$

where $\cos^{-1}(\angle v_1 v_2 v_3)$ is determined using the law of cosines in $\triangle v_1 v_2 v_3$. The angle can also be calculated using the Euclidean distances between vertices:

$$\Phi_{123} = \pi - \cos^{-1}\left[\frac{({e_{12}}^2 + {e_{23}}^2 - {e_{31}}^2)}{2 e_{12} e_{23}}\right] \tag{2.2}$$

where $e_{ij} = d(v_i, v_j)$ represents the Euclidean distance between vertices v_i and v_j.

To compute the Euclidean distance $D_{(u,v)}$ between two nodes u and v, the following formula is used:

$$D_{(u,v)} = \sqrt{(x_u - x_v)^2 + (y_u - y_v)^2} \tag{2.3}$$

where (x_u, y_u) and (x_v, y_v) are the coordinates of nodes u and v, respectively.

In UAV path planning, optimizing trajectory length and mission completion time is a critical aspect. The mission completion time accounts for both flight and hovering durations. A commonly used formula for mission completion time [41] is given by:

$$\tau = \frac{\top}{\upsilon} + \sum_{\vartheta=1}^{\mho} \frac{\Phi_\vartheta}{\varpi} \tag{2.4}$$

where $\top$ represents the path length, υ is the UAV speed, $\mho$ is the number of turns, Φ_ϑ is the angle of the ϑth turn, and ϖ denotes the UAV rotation rate.

The energy consumption model follows the approach in [42]. The energy cost $W(u,v)$ for traversing an edge between nodes u and v is:

$$W(u,v) = \lambda D_{(u,v)} \tag{2.5}$$

where λ denotes the energy consumption per unit distance, and $D_{(u,v)}$ is the Euclidean distance between the nodes. The energy required for making a turn $W(\Phi_{\widehat{uv'v}})$ is given by:

$$W(\Phi_{\widehat{uv'v}}) = \gamma \frac{180}{\pi} \Phi_{\widehat{uv'v}} \tag{2.6}$$

where γ represents the energy consumption per degree of turning angle. The values of these parameters should be selected based on the specific application and validated against values reported in the literature for comparison.

The total energy consumption E is expressed as the sum of distance-based and turn-based energy components:

$$\Xi = \sum_{u \in \mathscr{V}} \sum_{v \in \mathscr{V}} W(u,v) + \sum_{\text{turns}} W(\Phi_{\widehat{uv'v}}) \tag{2.7}$$

Another critical KPI is the quality of coverage (QoC), which evaluates the percentage of the area covered by the planned path, especially in NFZ avoidance scenarios. The QoC is defined as [42, 63, 64].:

$$QoC = \frac{CN}{TN} \times 100\% \tag{2.8}$$

where CN is the number of covered non-zero grid cells and TN is the total number of non-zero cells (excluding NFZ cells).

CHAPTER III

STATE OF THE ART

3.1 USAGE OF UAVS IN HEALTHCARE MISSIONS

3.1.1 OVERVIEW

Researchers have focused on the use of UAVs in healthcare. Some recent studies addressed the acceptability of UAVs in the health sector [65, 66] and ethical perspectives [67, 68]. Some studies compared the performance of UAVs sponsored by well-known firms for healthcare-related missions [65, 69]. One of the primary operations of UAVs in healthcare is service delivery due to their low cost compared to other vehicles.

Delivery missions include organ delivery [70, 71, 72, 73, 74], blood delivery [69, 75, 76, 77, 78], delivering small medical devices and medicines [79, 80, 81, 82, 83], and defibrillators [84, 85, 86, 87, 88, 89, 90]. UAVs also played an important role during the COVID-19 pandemic. They have been employed to monitor or deliver various types of goods, medical supplies, and test kits [91, 92, 93, 94, 95, 30, 96, 97]. Additionally, the use of UAVs has been explored in various approaches for SAR missions in healthcare [98, 99, 100, 101, 102]. Figure 3.1 illustrates the distribution of UAV use in different health sector domains.

The UAV delivery task in healthcare is one of the most critical missions [103] because of different concerns, limitations and challenges. This issue was the main theme for the works by [104, 105, 92, 28]. Details about these concerns will be presented in Section 3.1.3.

In any pandemic or disaster or in a rural setting that is not well connected to the health care system, UAVs can contribute to helping patients and supporting medical centres

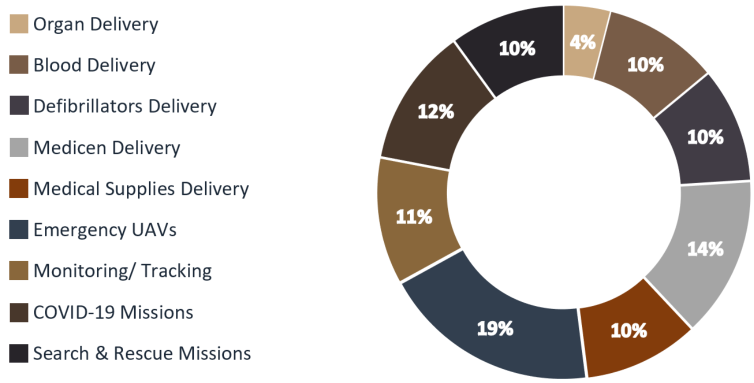


Figure 3.1 Distribution of UAV use in the healthcare sector, as reported in the findings of Section 3.1.4.

or laboratories by delivering medical supplies[106]. UAVs can operate in these situations because they are designed to fly over a medium range at reasonable speeds and carry sufficient payloads[107].

3.1.2 HEALTHCARE MISSIONS

DELIVERY OF MEDICAL SUPPLIES

One of the most crucial issues in healthcare is organ delivery. The organ shortage crisis puts many challenges on delivery missions. Surgeons report that using UAVs to assist with organ transplants could improve patient care [70]. However, the most challenging task is to ensure the safety of the organs being carried or delivered. For example, organs might be stolen during delivery. In addition, organs need specific treatment because they are sensitive to time and temperature. Figure 3.3 shows the time limit for different organs to remain viable outside of the body.

Figure 3.2 AED attached to the UAV (Life-UAV AED).

Other supplies have been delivered, such as pathology specimens, laboratory test samples, and HIV therapies [94, 76, 77, 83]. UAVs can also deliver supplies that are not typically considered urgent needs, such as maintenance drugs and contraceptives for women [108].

ORGAN DELIVERY

One of the most crucial issues in healthcare is organ delivery. The organ shortage crisis puts many challenges on delivery missions. Surgeons report that using UAVs to assist with organ transplants could improve patient care [70]. However, the most challenging task is to ensure the safety of the organs being carried or delivered. For example, organs might be stolen during delivery. In addition, organs need specific treatment because they are sensitive to time and temperature. Figure 3.3 shows the time limit for different organs to remain viable outside of the body.

Research and education are both needed before adopting an innovative organ transplant system using UAV systems [70]. Figure 3.4 shows an image taken while delivering a kidney in [70].

Figure 3.3 Organ survival outside the body.

Figure 3.4 Researchers from the University of Maryland proposed a study on delivering a kidney using a UAV.

BLOOD DELIVERY

UAVs could be used to deliver blood samples between laboratories and hospitals. They offer the advantages of low cost, careful transportation and shorter delivery time of life-saving blood samples. For example, when blood is delivered via trucks driving on a poorly maintained road with potholes and bumps, it bounces around and vibrates, causing the blood sample to be damaged [103]. Furthermore, blood is sensitive to temperature and must be transported within a specific temperature range. Traffic jams in cities and mega-cities may cause delays in delivery time when transporting these products. Many of these issues can be solved with UAVs.

UAVs are being tested in Southampton and other cities for various purposes, including transportation of blood samples for pathology labs [109] as shown in Figure 3.5. The trial of transportation of blood samples aims to help pathology labs to process samples more efficiently.

Figure 3.5 Transportation of blood samples using UAV [109].

In [78], a proof-of-concept pilot study was conducted at an urban trauma center to determine whether UAVs could deliver simulated blood products more rapidly than ground transportation and investigate their use in an urban environment. The study considered challenges such as blood product temperature stability, population safety, and legal parameters. Researchers in [110] propose a method for UAVs to transport blood from a central location to hospitals nearby in 15-20 minutes. In [111], the authors studied the problem of delivering blood supplies to multiple emergency sites using UAVs and proposed an optimal method for calculating the mixture of blood and transportation appendage in different conditions. In [112], the authors addressed the stability of biological samples in prolonged drone flights by obtaining paired samples from 21 adult volunteers. In [113], the authors examined the feasibility of drone transportation for blood products, finding simple solutions for maintaining target temperatures during flight and providing a template for future experiments.

UAVS AND COVID-19

According to the World Health Organization (WHO), more than 6.19 million individuals have died from the coronavirus [114]. Some people lacked support during the pandemic due to distance and safety protocols; it was difficult to deliver assistance, whether it was meals or medicine [115]. Even medical staff could not service all of the patients in some circumstances. UAVs were implemented in this case to provide assistance [91]. UAVs and robots have been used in Africa for healthcare purposes [92]. The outbreak of the pandemic revolutionized health services in Africa and signalled a shift in the use of UAVs in healthcare [92]. Due to many factors, such as deteriorating road networks, weak health systems, insufficient health funding, poor network connectivity, especially in rural areas, and outbreaks of infectious diseases, the increased use of UAV technology in health care is inevitable [92].

COVID-19 forced researchers and medical firms to develop innovative healthcare delivery systems [96, 97] using concepts that might have been unthinkable even 2 years ago. The use of UAVs for medical services is now being tested or implemented [103]. In certain places, such as parts of China, UAVs were utilized manually to monitor and track the movement of people in the streets and identify those not wearing masks, as shown in Figure 3.6.

Authors in [62] proposed a scheme that examined how UAVs played a critical part in helping individuals in quarantine during the COVID-19 in Wuhan, China, including monitoring post-epidemic damage and other activities. UAVs were also deployed to spray sanitizers in public areas.

In some places, UAVs were used to deliver COVID-19 test samples, such as PCR tests. In the work of [116], a Zipline UAV was implemented to transport COVID-19 sample tests in Ghana, as shown in Figure 3.7; the contribution of that project was to deliver samples between

Figure 3.6 Using UAV technology to combat COVID-19 in China

rural health areas and laboratories. The samples were delivered using packages sent to and received from the tops of plastic stands placed outside laboratories and hospitals. The paper outlined a method for delivering medical supplies. However, the problem of maximizing the UAV delivery range was not solved, and the entire process was not completely automated, necessitating the hiring of additional employees to assist in the sending and receiving of parcels.

Figure 3.7 Zipline UAV in Ghana transporting COVID-19 test samples.

The samples were delivered using packages sent to and received from the tops of plastic stands placed outside laboratories and hospitals. The paper outlined a method for delivering medical supplies. However, the problem of maximizing the UAV delivery range was not solved, and the entire process was not completely automated, necessitating the hiring of additional employees to assist in the sending and receiving of parcels.

UAVs have been used to deliver self-testing kits to individuals who request them. The study of [30] focused on efforts to tackle COVID-19 in Pakistan using AI-powered UAVs. The authors mentioned the COVID-19 sample collection techniques and self-testing kits for patients to use at home. The paper also compared the firms and markets that offer these kits. An example is shown in Figure 3.8.

Figure 3.8 Delivering COVID-19 kits.

SEARCH AND RESCUE

Search and Rescue (SAR) missions in healthcare are critical operations that involve the immediate and organized response to emergency situations. Specialized teams of healthcare professionals, including paramedics, emergency medical technicians (EMTs), and doctors, carry out these missions. These teams are equipped with the necessary medical equipment and supplies to provide life-saving care to patients in need. The primary goal of SAR missions is to quickly locate and extract patients from dangerous or inaccessible areas, and transport them to a medical facility for further treatment. This may include providing advanced life support, stabilizing injuries, and ensuring the safe transport of patients to a hospital or other medical facility. In order to ensure the success of SAR missions, it is essential for healthcare organizations to have well-trained and equipped teams in place, as well as clear protocols for communication and coordination among team members.

The authors in [99] focus on using UAVs for search and rescue missions in emergency scenarios, specifically to locate and aid injured civilians. The mission is split into two phases: the first phase involves using UAVs to scan designated areas and detect individuals in need of assistance, and the second phase involves delivering necessary medical supplies and aid to those identified. In [100], a UAV-based system utilizing a fleet of specialized UAVs is proposed for search and rescue operations in disaster scenarios. In [101], the authors propose two construction heuristics and a metaheuristic approach to improve the initial solution for vehicle routing problems. In [102], the authors present a solution for using multiple Micro Aerial Vehicles (MAVs) for cooperative search and rescue in post-disaster situations, discussing the system configuration and key technologies developed for the mission.

Mainly, UAVs are being used in healthcare in different missions. UAVs in these missions face challenges that include regulatory barriers, and safety concerns. It is crucial that UAVs navigate efficiently in order to achieve these missions. The following section discusses the challenges UAVs face in their missions, particularly in healthcare.

3.1.3 UAV CHALLENGES IN HEALTHCARE

The integration of UAVs into healthcare missions also presents a number of challenges that must be overcome. One major challenge is ensuring the safety of both the UAVs and the people they interact with while achieving the mission goal. UAVs must be able to navigate safely in crowded and complex environments, such as urban areas or in the vicinity of hospitals and other medical facilities, to avoid collisions with other aircraft and obstacles. Additionally, UAVs must be able to operate reliably and withstand extreme weather conditions, such as high winds, heavy rains, and hot temperatures, to ensure that they can carry out their missions without fail. Another challenge is ensuring the security and privacy of sensitive medical data and personal information that may be transported by the UAVs. UAVs must be equipped with robust security measures to protect against hacking and other forms of cyberattacks, and they must comply with all relevant regulations and laws regarding data privacy and protection.

Moreover, concerns and regulations to fly UAV are another major challenge. UAVs must operate within the framework of national and international regulations, such as those issued by the Federal Aviation Administration and the International Civil Aviation Organization, to ensure that they are operated safely and legally. These regulations may impose strict limitations on where and when UAVs can fly, which can make it difficult to carry out certain healthcare missions. The healthcare system must adapt to local circumstances to successfully implement UAVs [18]. Time plays a major role in most healthcare missions, and battery life is a major challenge for UAVs. Other challenges must be addressed. For example, while delivering an organ or blood samples, in addition to travel duration and battery life, the organ temperature and the UAV stability should be considered.

Other challenges arise in medical supplies delivery, such as payload and parcel capacity. As for the COVID-19-related missions, a fleet of UAVs could be used to monitor and track people, bringing more challenges, such as network connectivity, collision avoidance, area coverage and communication between the UAVs. A path must be planned for the UAV in any mission, and in any path planning problem, a set of challenges must be considered to provide a realistic representation of the real world and the difficulties that could arise. Realistic challenges must be considered to present a feasible solution. Reducing mission time, saving UAV energy and obstacle avoidance are major challenges. Failure to address these challenges could cancel a mission if a path cannot adapt to sudden events. Time and energy are influenced by speed, distance, UAV rotation rate, turning angles, and other factors.

Researchers who provide solutions for healthcare missions utilizing UAVs must be aware of the challenges that could occur during these missions. The main challenges considered in most recent work that tackles path planning problems are lowering energy consumption, minimizing mission completion time [41, 53, 42], wind factors [117, 118], obstacle avoidance and many other challenges [53]. There are two types of obstacles: static obstacles like trees, buildings, or signboards [53] (Figure 3.9a,3.9b) and dynamic obstacles such as birds, balls, or another UAV [119] (Figure 3.9c,3.9d).

(a) UAV facing trees. (b) UAV stuck in a tree. (c) UAV avoiding a bird. (d) UAV chased by a bird.

Figure 3.9 UAV facing static and dynamic obstacles.

Other concerns in any mission include maximizing coverage range [120], optimizing paths [121], and ensuring smooth path planning [122]. Additionally, when using a fleet of UAVs, ensuring connectivity [123] between the UAVs and avoiding collisions are critical challenges. One possible approach is to divide an area into sub-areas and assign a UAV to each sub-area, as proposed in the research by [42]. Partitioning helps eliminate UAV collisions and prevents UAVs from flying over non-flying zones (NFZ). For reasons of safety or privacy, these NFZs are areas where UAVs are not allowed to fly or pass over.

A live saving mission implement critical challenges to the path planning problem in healthcare missions. Despite the fact that UAVs can't handle all challenges in a mission, we highlighted the most common and difficult challenges in the literature, which provided algorithms and techniques for planning safe paths for UAVs.

3.1.4 REVIEW AND ANALYSIS

A comparative analysis of various existing approaches and case studies using UAVs in healthcare missions is shown in Table 3.1. Mission completion time is one of the main constraints in any mission. Table 3.1 show that 93.48% of the proposed approaches and case studies evaluated their work with respect to time. Of these approaches, 58.70% taken into account path length as performance metric.

However, only 26% of the approaches were aware of energy consumption. This reflects a lack of energy-aware paths and solutions for UAVs in healthcare missions. Additionally, mission cost is mentioned in only 17.40% of the studies, which indicates that most of the works did not take mission cost as a priority.

Planning a path for certain delivery missions in health (organ delivery, patient rescue) must also consider important factors, such as the state of the delivered product or the rescued patient. For instance, the parcel temperature factor is mainly referred to in organ, blood, and medicine delivery missions. The heart-rate factor is referred to missions related to emergency and first aid delivery where the UAV could monitor and track the patient's heart rate. Such factors may change the whole planned path. Temperature was mentioned in 24% of the works, and only 6.52% took patient heart rate into account.

Figure 3.10 shows that 35% of the papers tested their works in real life, but 50% of those did not include information about the flight path. A simulator was used in about 46% of the works, but 33% of the studies that used simulators did not incorporate the flight path. About 6% of approaches tested their works in both simulator and in real-life where they provided information on both path planning algorithms and flight paths. In 13% of the studies, in 83.33% out of these works there was no information about testing or results or mention of the flight path. These studies raise questions about how UAVs would act in the real world.

Almost 85% of the approaches neglected the wind effect, which is one of the most important effects in real-life scenarios. Obstacle avoidance challenges must also be considered to allow a realistic integration of UAVs in healthcare missions. Researchers must be aware of the environmental factors that surround the UAV in any healthcare mission. Safe paths are among the most important factors; parcels must also be protected from the elements and from diversion.

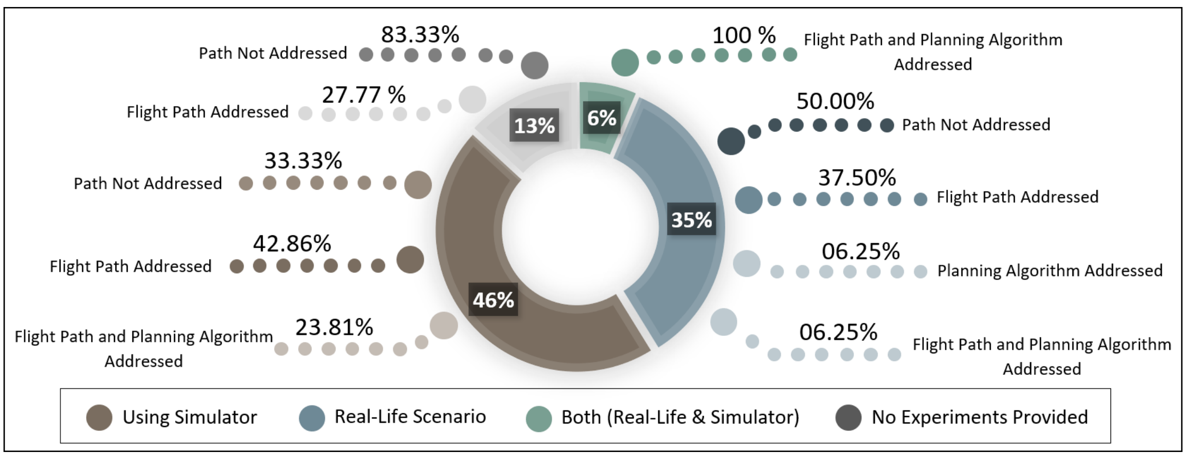


Figure 3.10 Overview of approaches and studies that addressed path planning and experimental conditions.

Table 3.1 Comparison of 46 related works and case studies to healthcare missions.

Case Studies References		[3]	[25]	[26]	[27]	[30]	[73]	[74]	[78]	[79]	[84]	[85]	[86]	[87]	[88]	[89]
Mission Type	Organ delivery	-	✓	-	-	-	✓	-	-	-	-	-	-	-	-	-
	Blood delivery	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-
	Defibrillator delivery	-	-	-	-	-	-	-	-	-	✓	✓	✓	✓	✓	✓
	Medicine delivery	-	-	✓	-	-	-	-	-	✓	-	-	-	-	-	-
	Medical support delivery	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Emergency UAV	✓	-	✓	✓	-	-	✓	-	-	-	-	-	-	-	-
	Patient monitoring	-	-	✓	✓	-	-	-	-	-	-	-	-	-	-	-
	Self-testing kit delivery	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-
	Spraying sanitizers	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Search and rescue	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
COVID-19	Related to pandemic	-	-	✓	-	✓	-	-	-	-	-	-	-	-	-	-
Main Challenges	Completion time	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Mission cost	-	-	✓	-	✓	-	-	-	-	-	-	-	-	-	-
	Energy consumption	-	-	✓	-	-	✓	-	-	-	-	✓	✓	-	-	-
Factors/Metrics	Wind effect	-	-	-	-	-	-	-	✓	-	✓	✓	-	-	✓	-
	Speed/acceleration	✓	-	-	-	-	-	-	✓	✓	✓	✓	✓	✓	✓	✓
	Covered distance	✓	-	-	-	✓	-	-	-	✓	✓	✓	-	-	✓	✓
	Payload size	✓	-	-	-	-	-	-	✓	✓	-	✓	-	-	✓	-
Special Challenges	Parcel temperature	-	✓	-	-	-	-	-	✓	✓	-	-	-	-	-	-
	Patient heart rate	-	-	-	✓	-	-	✓	-	-	-	-	-	-	-	-
Path Planning	Path is not addressed	-	✓	-	-	-	✓	-	✓	-	✓	-	✓	-	✓	-
	Flight path explained	✓	-	✓	-	✓	-	✓	-	✓	-	✓	-	✓	-	✓
	Path planning algorithm	-	-	✓	✓	✓	-	-	-	-	-	-	-	-	-	-
Testing and Results	Real life	-	-	-	✓	-	-	✓	✓	✓	✓	-	-	-	✓	-
	Using simulator	✓	✓	✓	-	✓	✓	-	-	-	-	✓	✓	✓	-	✓

✓ Check mark symbol indicates the feature presence. - Dash symbol indicates the absence of a feature.

Continue Table 3.1: Comparison of 46 related works and case studies to healthcare missions.

Case Studies References		[90]	[96]	[97]	[98]	[99]	[100]	[101]	[102]	[110]	[111]	[112]	[113]	[124]	[125]	[126]
Mission Type	Organ delivery	-	-	-	-	-	-	-	-	-	-	-	-	-	✓	-
	Blood delivery	-	-	-	-	-	-	-	-	✓	✓	✓	✓	-	-	-
	Defibrillator delivery	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Medicine delivery	-	-	-	-	-	-	-	-	-	-	-	-	-	-	✓
	Medical support delivery	-	-	-	-	-	-	-	-	-	-	-	-	✓	-	-
	Emergency UAVs	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-
	Patient monitoring	-	✓	✓	✓	-	-	-	-	-	-	-	-	-	-	-
	Self-testing kit delivery	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Spraying sanitizers	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-
	Search and rescue	-	-	-	✓	✓	✓	✓	✓	-	-	-	-	-	-	-
COVID-19	Related to pandemic	-	✓	✓	-	-	-	-	-	-	-	-	-	-	-	-
Main Challenges	Completion time	✓	✓	✓	✓	✓	-	✓	-	✓	✓	✓	✓	✓	✓	✓
	Mission cost	✓	-	-	-	-	-	-	-	-	-	-	-	✓	-	-
	Energy consumption	-	✓	-	-	-	✓	✓	-	✓	-	-	-	-	-	-
Factors/Metrics	Wind effect	-	-	-	✓	-	-	-	-	-	-	-	-	-	-	-
	Speed/acceleration	✓	✓	✓	-	-	✓	-	-	-	-	-	-	✓	-	-
	Covered distance	✓	✓	✓	✓	✓	-	-	✓	✓	✓	-	-	✓	-	✓
	Payload size	-	-	✓	-	-	-	-	-	✓	✓	-	-	-	-	-
Special Challenges	Parcel temperature	-	-	-	-	✓	-	-	-	✓	✓	✓	✓	-	✓	-
	Patient heart rate	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Path Planning	Path is not addressed	-	✓	-	✓	-	-	-	-	✓	-	✓	✓	-	✓	-
	Flight path explained	✓	-	✓	-	✓	✓	✓	✓	-	✓	-	-	✓	-	✓
	Path planning algorithm	-	-	✓	-	✓	-	✓	✓	-	✓	-	-	-	-	-
Testing and Results	Real life	✓	-	-	✓	✓	-	✓	✓	-	✓	✓	✓	-	-	-
	Using simulator	-	✓	✓	-	✓	-	✓	✓	-	-	-	-	✓	✓	✓

✓ Check mark symbol indicates the feature presence. - Dash symbol indicates the absence of a feature.

Continue Table 3.1: Comparison of 46 related works and case studies to healthcare missions.

Case Studies References		[127]	[128]	[129]	[130]	[131]	[132]	[133]	[134]	[135]	[136]	[137]	[138]	[139]	[140]	[141]	[142]
Mission Type	Organ delivery	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Blood delivery	-	-	-	-	-	-	-	-	-	-	-	-	✓	✓	-	-
	Defibrillator delivery	-	-	-	-	-	✓	-	-	-	-	-	-	-	-	-	-
	Medicine delivery	-	-	-	-	-	-	-	-	-	-	-	✓	✓	-	-	-
	Medical support delivery	-	-	-	-	✓	-	✓	-	-	✓	✓	✓	-	-	-	-
	Emergency UAVs	✓	-	-	✓	✓	✓	-	✓	✓	-	-	-	-	-	✓	-
	Patient monitoring	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	✓
	Self-testing kit delivery	-	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-
	Spraying sanitizers	-	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Search and rescue	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
COVID-19	Related to pandemic	-	✓	✓	-	-	-	✓	-	-	-	-	-	-	-	-	-
Main Challenges	Completion time	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Mission cost	✓	-	-	✓	-	-	-	-	-	-	-	-	-	✓	✓	-
	Energy consumption	✓	-	✓	-	-	✓	-	-	-	-	-	✓	-	-	-	-
Factors/Metrics	Wind effect	-	-	-	-	-	-	✓	-	-	-	-	-	✓	-	-	-
	Speed/acceleration	✓	✓	-	-	✓	✓	✓	-	✓	✓	-	✓	✓	-	✓	-
	Covered distance	✓	✓	✓	✓	-	-	✓	-	✓	-	-	✓	✓	✓	✓	-
	Payload size	✓	-	✓	-	✓	✓	✓	-	-	-	✓	✓	✓	-	-	✓
Special Challenges	Parcel temperature	-	-	✓	-	-	-	-	-	-	-	-	-	✓	-	-	-
	Patient heart rate	-	-	-	-	-	-	-	✓	-	-	-	-	-	-	-	-
Path Planning	Path is not addressed	-	-	-	-	✓	✓	-	✓	-	✓	-	-	✓	✓	✓	✓
	Flight path explained	✓	✓	✓	✓	-	-	✓	-	✓	-	✓	✓	-	-	-	-
	Path planning algorithm	-	✓	-	✓	-	-	-	-	-	-	-	-	-	-	-	-
Testing and Results	Real life	-	-	✓	-	-	-	✓	-	-	-	✓	-	✓	✓	-	-
	Using simulator	✓	✓	-	✓	-	-	-	-	✓	-	-	✓	-	-	✓	✓

✓ Check mark symbol indicates the feature presence. - Dash symbol indicates the absence of a feature.

3.2 OBSTACLE DETECTION AND AVOIDANCE (ODA) METHODS

3.2.1 OVERVIEW

The safety of the UAV and its mobility need to be controlled while monitoring areas of interest. A planning technique must control the UAV along a collision-free path [143, 144, 145, 146], especially in the presence of multiple UAVs [147, 148, 149]. In addition, for a UAV mission to be successful, the application domain restrictions (obstacles and NFZ) must be addressed in the planning strategy [150]. NFZs are zones over which aircraft are not allowed to fly [42]. In several works [42, 63, 64, 151, 52], NFZs are viewed as static obstacles in a two-dimensional environment. One of the main challenges in a mission is to find an optimal path that completely covers an area of interest and avoids obstacles or NFZs without consuming excessive energy or time [41].

The obstacle detection and avoidance (ODA) methods in this chapter are categorized according to the avoidance techniques. An overview of the ODA mission components is represented in Figure 3.11. In addition, a comparison analysis of the mentioned case studies is presented based on the main mission components.

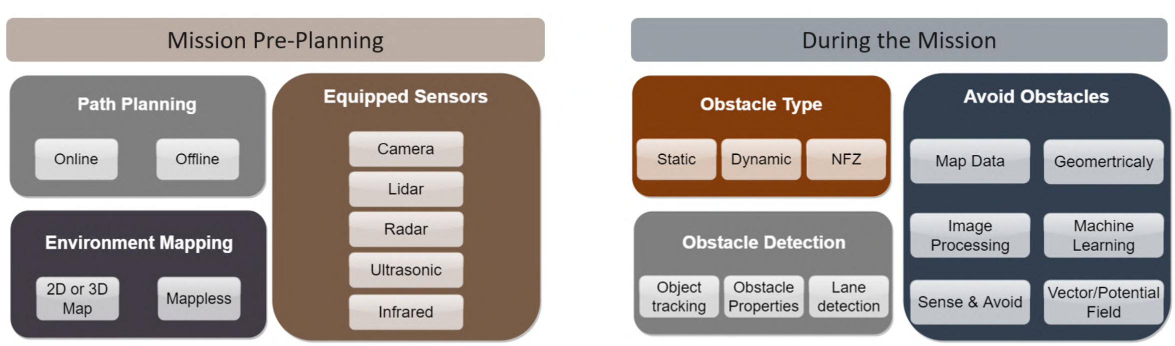


Figure 3.11 ODA Mission Components Overview

3.2.2 ENVIRONMENT MAPPING

Environment mapping is crucial for UAV path planning, providing visual data to ensure safe navigation [152]. Mapping frameworks can be either 2D or 3D, depending on the mission requirements. These maps are either pre-generated before the mission or created in real-time using UAV sensors. Table 3.2 summarizes the key mapping techniques, including their advantages and limitations. By selecting the appropriate mapping technique, UAV systems can enhance their ability to navigate complex environments while maintaining safe and efficient paths.

Table 3.2 Comparison of Mapping Techniques for UAV Path Planning

Mapping Technique	**Advantages**	**Limitations**
Depth Map [153]	Provides direct distance information, useful for detecting obstacles and localizing objects.	Sensitive to lighting conditions; limited range in low-contrast regions.
OctoMap [154, 155]	Efficient 3D representation, supports volumetric mapping, ideal for large-scale environments.	Computationally expensive; requires significant memory for high-resolution maps.
RTAB-Map [156, 157, 158]	Real-time 3D SLAM with visual and LiDAR data integration; handles loop closure detection.	May struggle with rapid UAV motion and sensor noise in dynamic environments.
Occupancy Grid [159, 160, 161]	Simple and computationally efficient; effective for 2D navigation and obstacle avoidance.	Limited to 2D environments; cannot represent complex 3D structures.
Disparity Map [162]	Useful for depth estimation from stereo images; effective for real-time applications.	Errors in depth estimation for long distant or low-texture objects.
UV Map [163]	Enhances visual detail on 3D models; useful for texture mapping in virtual simulations.	Not directly useful for path planning; primarily for visualization.

Depth maps encode object distances as image intensities, providing crucial data for obstacle detection and localization. This information is essential for UAVs to understand their surroundings and avoid collisions, as illustrated in Figure 3.12a [153]. Complementing depth maps, OctoMaps utilize a 3D occupancy grid to represent large-scale environments, enabling efficient volumetric mapping and path planning. This approach is particularly useful for UAVs operating in complex, unstructured spaces (see Figure 3.12b) [154, 155, 164].

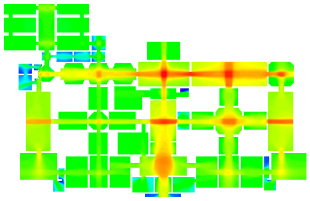
In real-time navigation tasks, RTAB-Map combines visual and LiDAR data for 3D simultaneous localization and mapping (SLAM). This integration allows for loop closure detection, which mitigates cumulative drift errors and enhances positional accuracy, as depicted in Figure 3.12c [156, 157, 158]. For simpler environments, Occupancy Grid maps offer a more straightforward representation by categorizing areas as occupied, free, or unknown on a 2D grid. This method is advantageous for basic path planning tasks (see Figure 3.12d) [159, 160, 161].

While UV maps primarily apply 2D textures to 3D models to improve visualization, they do not directly aid UAV navigation (shown in Figure 3.12e) [163]. In contrast, Disparity maps play a crucial role in depth estimation by measuring pixel displacement between stereo image pairs. This technique allows UAVs to distinguish object proximity, where closer objects appear brighter and distant ones darker, as illustrated in Figure 3.13 [162].

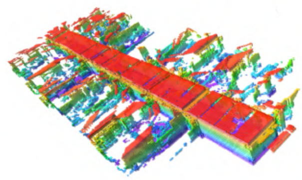
3.2.3 SENSORS USED FOR OBSTACLE DETECTION

UAVs rely on a variety of sensors for obstacle detection, mapping, and navigation. These sensors differ in their capabilities, costs, and limitations. To improve accuracy UAVs often integrate multiple sensors known as sensor fusion.

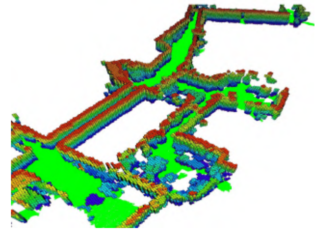
Among the most commonly used sensors are camera-based systems, including stereo, monocular, RGB-D, and first-person view (FPV) cameras. Stereo cameras provide 3D depth perception by analyzing two images, but their performance deteriorates in low-light conditions [165, 166, 167, 168, 169, 170, 171, 172, 173, 174]. Monocular cameras, though cost-effective,



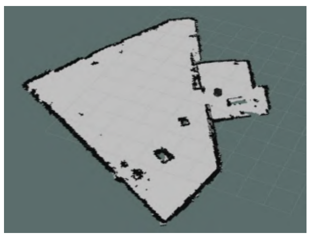
(a) Depth map [153]



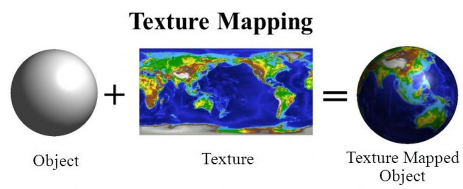
(b) OctoMap [164]



(c) RTAB-Map [156]

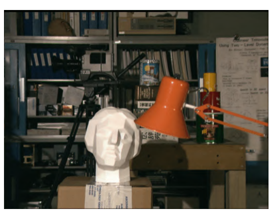


(d) Occupancy Grid Map [164]



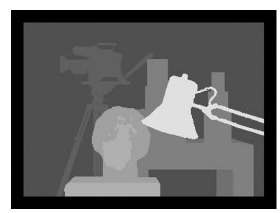
(e) UV map [163]

Figure 3.12 Examples of mapping techniques for UAV path planning.



(a) Left Camera

(b) Right Camera



(c) Disparity Image

Figure 3.13 Disparity map creation from stereo images [162].

capture only 2D images and rely on image processing for depth estimation [53, 154, 175, 176, 177, 178, 179, 119]. RGB-D cameras combine color and depth information, offering a balance between cost and perception capability [158, 180, 181, 157]. FPV cameras provide real-time video feeds and are lightweight, making them suitable for navigation but susceptible to visibility issues in poor lighting [182, 183, 184, 185, 186, 161].

Beyond cameras, other sensors enhance UAV performance under various environmental conditions. Lidar utilizes laser beams to generate highly accurate 3D maps, making it one of the most precise but also the most expensive solutions [170, 126]. MIMO radar operates using radio waves, making it unaffected by lighting conditions while also enabling direct velocity measurement through the Doppler effect [176, 187, 183]. Ultrasonic sensors, which rely on sound waves, are influenced more by environmental factors than by lighting conditions [188, 189, 190, 174, 191]. Infrared sensors detect heat radiation, but their effectiveness can be impacted by ambient infrared interference [190, 191].

When selecting sensors, it is crucial to consider their limitations. For instance, stereo cameras struggle with long-range depth estimation, monocular cameras lack depth perception, and some cameras fail to detect transparent or reflective obstacles [176]. While lidars are highly effective, they come at a significant cost. MIMO radar is robust against environmental interference but requires complex processing. To overcome these weaknesses, UAVs often use sensor fusion, combining multiple sensors to enhance reliability [192].

Ambient light significantly affects sensor performance. Cameras, including stereo, monocular, RGB-D, and FPV types, perform poorly in low-light conditions [193, 194, 195, 196]. However, lidar and MIMO radar remain unaffected, making them ideal for night-time or low-visibility operations. Ultrasonic and infrared sensors behave differently—ultrasonic performance depends on environmental factors, while infrared is sensitive to external heat sources.

Velocity estimation is another critical aspect of UAV navigation. Stereo cameras can estimate velocity by analyzing depth variations across frames, whereas monocular cameras only provide relative velocity estimates [197, 198]. Lidar enables highly accurate velocity measurements based on distance calculations, while MIMO radar directly measures velocity using the Doppler effect. Ultrasonic and infrared sensors are generally ineffective for precise velocity estimation.

Table 3.3 summarizes how different sensors perform under varying ambient light conditions and their velocity estimation capabilities.

Table 3.3 Sensor Performance Under Ambient Light and Velocity Estimation Capabilities

Sensor	**Impact of Ambient Light**	**Velocity Estimation**
Stereo Camera	Reduced depth perception in low light	Estimates depth-based velocity
Monocular Camera	Image quality may degrade in dim conditions	Provides relative velocity only
RGB-D Camera	Depth accuracy can degrade in low light	Limited velocity estimation
FPV Camera	Visibility can be hindered in low light	No significant velocity estimation
Lidar	Unaffected by ambient light	Highly accurate distance-based velocity
MIMO Radar	Unaffected by ambient light	Direct velocity measurement via Doppler effect
Ultrasonic Sensor	Less affected by light, depends on environment	Poor velocity estimation
Infrared Sensor	Sensitive to ambient infrared radiation	Ineffective for velocity estimation

3.2.4 STATIC & DYNAMIC OBSTACLE AVOIDANCE METHODS

In path planning or monitoring missions obstacles could pop up or interrupt the UAV's path. As mentioned previously obstacles could be static or dynamic and could be detected using different types of sensors. After detecting these obstacles various methods and techniques could be designed to plan safe paths for UAVs to prevent mission failure and to assure UAV safety.

The avoidance techniques and methods are categorized as follows: Avoidance using map data, geometric-based, using image processing, machine learning, sensor data, vector field, or potential field. Approaches could make use of one method or can combine two or more methods depending on the mission components.

MAP DATA

Mapping is fundamental for UAV navigation, whether pre-generated or built dynamically during flight. The integration of 2D and 3D map data significantly improves path planning and obstacle avoidance. Various studies explore different mapping techniques, each with distinct advantages and limitations.

Grid-based methods provide structured path planning. The Energy-Aware Obstacle Avoidance (EAOA) algorithm [53] initially generates a 2D path but refines it using a 3D frontal view when obstacles are detected. Similarly, [119] integrates A* for smoother, energy-aware UAV paths. However, offline approaches like [199] rely on predefined obstacle modeling and lack adaptability to dynamic environments.

Octomap-based techniques enable 3D spatial representation for obstacle avoidance. A bio-inspired plant growth algorithm [167] optimizes UAV movement within an Octomap, while [170] combines stereo cameras and LiDAR to construct probabilistic 3D maps. The 3VFH+ algorithm [155] extends 2D VFH+ to 3D, extracting obstacle features for optimal avoidance paths. Despite their accuracy, these methods can be computationally demanding.

Vision-based approaches, such as [165], use stereo cameras for disparity maps, while [200] employs an FPV camera and IMU for visual-inertial navigation (VINS). RGB-D cameras in [201] track obstacles in real time using Kalman filtering. These methods offer quick responses but may falter in poor lighting.

SLAM-based approaches excel in GPS-denied environments. RTAB-Map with a ZED camera [158] processes sensory data to build an adaptive 3D map, allowing UAVs to navigate static obstacles. Multi-sensor fusion in [157] enhances indoor UAV navigation in rescue missions. A feature-based planner [202] uses point clouds to classify obstacles and optimize real-time trajectory tracking.

Each method has strengths and limitations, summarized in Table 3.4:

Method	**Strengths**	**Weaknesses**	**Case Study**
Grid-based	Simple, structured, energy-efficient	Limited real-time adaptability	[53, 119, 199]
Octomap	Detailed 3D mapping	High computational cost	[167, 170, 155]
Vision-based	Real-time response	Sensitive to lighting	[165, 200, 201]
SLAM-based	Works in GPS-denied areas	Sensor fusion complexity	[158, 157, 202]

Table 3.4 Comparison of UAV Mapping and Obstacle Avoidance Methods

GEOMETRIC-BASED

Geometric-based approaches use spatial representations and mathematical models to ensure UAV obstacle avoidance and collision-free navigation. These methods vary from simple safety zones to advanced velocity and collision modeling techniques.

The 3D-SWAP algorithm [171] ensures safe UAV movement in swarms by modeling obstacles and UAVs as cylinders. Similarly, safety-zone approaches [182, 203] define circular safe areas around obstacles to prevent collisions. These methods are computationally efficient but struggle with dynamic threats.

Velocity obstacle methods [204, 205] predict potential collisions using obstacle velocity and position, selecting safe paths heuristically. While effective, they struggle with multiple moving obstacles without additional data. Collision cone methods [206, 207] generalize this by computing a range of heading angles to ensure collision-free navigation at constant speed, offering more adaptability.

Avoidance maps [208, 209] partition UAV control inputs into safe and collision-prone areas, efficiently guiding maneuvers. Meanwhile, binocular vision [169, 210] enhances obstacle detection by constructing disparity maps for depth perception, though it is sensitive to environmental conditions.

A hierarchical task-based approach [211] integrates graph theory to enable UAV formations while avoiding obstacles, balancing coordination and obstacle avoidance but requiring high computational resources.

Each method has strengths and limitations, summarized in Table 3.5:

Method	Strengths	Weaknesses	Case Study
3D-SWAP	Low computation, swarm-friendly	Assumes adequate braking distance	[171]
Safety Zones	Simple, effective for static obstacles	Limited for dynamic threats	[182, 203]
Velocity Obstacles	Fast, heuristic-based avoidance	Struggles with multiple dynamic obs.	[204, 205]
Collision Cone	Maintains constant speed	Requires precise obstacle modeling	[206, 207]
Avoidance Maps	Efficient, scalable	Control-space dependent	[208, 209]
Binocular Vision	Accurate depth estimation	Sensitive to lighting	[169, 210]
Hierarchical Task-Based	Effective in formations	High computational complexity	[211]

Table 3.5 Comparison of Geometric-Based UAV Avoidance Methods

IMAGE PROCESSING

Image processing techniques enhance UAV obstacle avoidance by analyzing visual data to detect, track, and navigate around obstacles. These methods range from monocular vision and depth estimation to radar-based detection, each addressing specific challenges in obstacle identification.

Monocular vision-based techniques allow UAVs to detect obstacles using feature-matching and template-based analysis. For example, the method in [177] utilizes Speeded-Up Robust Features (SURF [212]) and template matching to compare obstacle sizes across different frames, distinguishing near and distant objects even in complex backgrounds. However, monocular approaches often struggle with depth estimation in low-texture environments, necessitating additional strategies.

To improve upon this limitation, a size expansion-based approach [175] mimics human perception by analyzing obstacle size changes as the UAV moves closer. This enables real-time collision probability estimation, allowing the UAV to adjust its trajectory dynamically. While effective within a range of 90–120 cm, this approach is constrained by distance limitations and may require supplementary sensors for broader coverage.

For improved accuracy, radar-based image processing [183] incorporates video frame analysis, capturing multiple sequential images to detect obstacles. By applying background subtraction and edge detection, this method accurately estimates object size and clearance distance, ensuring a well-defined avoidance path. Although effective, it relies on multiple frames, which may introduce delays in fast-moving scenarios.

To address real-time responsiveness, stereo vision methods [213] leverage depth images to construct a precise 3D obstacle map. Unlike monocular approaches, stereo cameras provide direct depth perception, enhancing obstacle detection accuracy. This information is processed and transmitted to the UAV's flight controller, dynamically adjusting its global path. However, stereo vision systems are susceptible to variations in lighting conditions, which can affect depth estimation reliability.

The integration of these methods can create a more robust obstacle avoidance system. While monocular vision offers a lightweight solution, combining it with depth-based or radar-assisted techniques improves reliability and accuracy. Future UAVs may benefit from hybrid approaches, optimizing real-time decision-making under diverse environmental conditions. Each method has strengths and limitations, summarized in Table 3.6:

Method	Strengths	Weaknesses	Case Study
Monocular Vision	Simple, lightweight	Limited depth perception	[177]
Size Expansion	Real-time collision prediction	Effective within short range	[175]
Radar-Based	Accurate object size estimation	Requires multiple frames	[183]
Stereo Vision	Precise depth perception	Sensitive to lighting variations	[213]

Table 3.6 Comparison of Image Processing-Based UAV Avoidance Methods

MACHINE LEARNING

Machine learning (ML) enhances UAV obstacle avoidance by enabling adaptive decision-making in complex environments. Various ML techniques, including neural networks, deep reinforcement learning, and object detection models, have been integrated to improve UAV navigation and collision avoidance.

Neural networks play a key role in learning-based avoidance strategies. In [178], a UAV learns obstacle avoidance by imitating human behavior using deep neural networks trained in a simulated environment using Gazebo [214]. However, such approaches rely heavily on extensive training data and may not generalize well to unseen scenarios. To address this, transformer-based learning [215] processes observation sequences to improve UAV decision-making in dynamic environments, effectively balancing tracking and avoidance.

Convolutional Neural Networks (CNNs) facilitate real-time object detection. Faster R-CNN [179] enables UAVs to detect and navigate around tree trunks using a single monocular camera, estimating obstacle-free paths based on detected tree spacing. Meanwhile, YOLO (You Only Look Once) is a real-time object detection algorithm that uses CNN to predict bounding box positions and class probabilities from input images [216]. YOLO-based approaches [217, 218] improve detection speed and accuracy, extending UAV perception to thermal infrared imagery for enhanced obstacle identification.

Reinforcement learning techniques enable UAVs to autonomously adapt to dynamic obstacles. Deep Q-Learning (DQL) [172] allows UAVs to learn obstacle avoidance policies in 3D environments by maximizing flight efficiency while minimizing collisions. An extension [219] incorporates priority sampling and exponential weighting to stabilize learning and improve decision-making under uncertainty. Additionally, a Q-learning-based framework [220] integrates a shortest-distance prioritization policy to optimize both local avoidance and global path efficiency.

Sensor fusion further enhances learning-based navigation. RealSense-based reinforcement learning [166] utilizes depth images to compute avoidance maneuvers, while integrated sensing and communication (ISAC) [221] improves UAV path-following using a virtual leader approach. Furthermore, a CNN-based end-to-end learning system [222] combines multiple convolutional techniques to predict collision probabilities and adjust UAV movement in real-time.

These methods collectively contribute to more robust UAV obstacle avoidance. Table 3.7 compares key learning-based techniques in terms of strengths, weaknesses, and applications.

Method	**Strengths**	**Weaknesses**	**Case Study**
Neural Networks	Human-like learning	Data-intensive training	[178, 215]
CNN-Based	Fast object detection	Limited to trained objects	[179, 217, 218]
Reinforcement Learning	Adaptive to dynamic obstacles	Requires exploration	[172, 219, 220]
Sensor Fusion	Enhanced perception	Computationally demanding	[166, 221, 222]

Table 3.7 Comparison of Machine Learning-Based UAV Avoidance Methods

SENSE & AVOID

Sense-and-avoid techniques combine various sensing modalities to improve UAV obstacle detection and avoidance. By integrating vision, radar, and ultrasonic sensors, these approaches enhance UAV navigation in dynamic environments.

Vision-based sensing provides critical spatial awareness. A system [176] fuses monocular camera data with millimeter-wave radar to extract 3D obstacle coordinates using an Extended Kalman Filter (EKF), enabling Bi-RRT* path planning. To enhance UAV swarm navigation, an active sensing framework [161] uses Gaussian mixture models (GMM) and distributed particle swarm optimization (DPSO) to generate a global consensus map, optimizing flight formations while ensuring collision avoidance.

Sensor fusion methods further improve obstacle tracking. An approach [180] integrates IMU and RGB-D camera data, using an Information Filter to predict obstacle motion and define escape points for collision-free movement. Similarly, a model predictive control (MPC)-based algorithm [223] employs a Bin-Occupancy filter to track and predict obstacles beyond the UAV's immediate field of view, refining navigation in cluttered indoor spaces.

Radar-based detection provides robust obstacle awareness. A 3D-sensing MIMO radar [187] alerts UAVs to moving obstacles within their flight path, triggering evasive maneuvers. Ultrasonic sensors [189] complement these techniques by fusing multiple sensor readings to detect objects in multiple directions, addressing limitations of single-sensor approaches. A low-cost system [190] combines ultrasonic and infrared sensors to refine 3D obstacle detection, ensuring efficient collision avoidance in resource-constrained UAVs.

For enhanced precision, stereo vision is paired with ultrasonic sensors [174], enabling UAVs to characterize their surroundings more accurately. By detecting obstacles and applying a rapidly exploring random tree (RRT) algorithm, this method improves UAV path planning in complex environments.

These methods collectively contribute to more robust UAV obstacle avoidance. Table 3.8 compares sense and avoid based techniques in terms of strengths, weaknesses, and applications.

Method	**Strengths**	**Weaknesses**	**Case Study**
Vision + Radar	Accurate 3D mapping	Requires sensor fusion	[176]
Swarm Sensing	Optimized formations	Computational complexity	[161]
IMU + RGB-D	Predictive obstacle tracking	Limited to known environments	[180, 223]
MIMO Radar	Effective for moving objects	Hardware-intensive	[187]
Ultrasonic Sensors	Low-cost, multi-directional	Limited range	[189, 190]
Stereo Vision + RRT	Precise depth estimation	Sensitive to lighting conditions	[174]

Table 3.8 Comparison of Sense-and-Avoid UAV Methods

VECTOR/POTENTIAL FIELD

Vector and artificial potential field (APF) methods are widely used in UAV obstacle avoidance and path planning. These approaches generate repulsive forces around obstacles and attractive forces towards goals, ensuring real-time adaptability in dynamic environments.

Traditional APF-based methods struggle with local minima, requiring enhancements to improve performance. A leader-follower APF strategy [224] increases formation flexibility, while an improved APF [225] incorporates Goal Not Reachable Near Obstacle (GNRON) modifications to handle multiple obstacles. Further optimization [226] introduces spatial and temporal constraints, enhancing avoidance speed and efficiency.

To overcome APF's limitations in dynamic environments, fuzzy logic-based improvements [227] integrate velocity repulsion fields, breaking local minima constraints and ensuring smoother avoidance paths. Similarly, a height-factor-enhanced APF [228] prevents UAV entrapment and maintains stability in fault conditions, while event-triggered APF [229] mitigates cyber-attack risks by treating compromised UAVs as obstacles.

Hybrid approaches combine APF with graph-based methods for improved navigation. A localized vector field and A*-based obstacle avoidance algorithm [160] plans paths dynamically by merging waypoints using the split and merge (SaM) algorithm. Similarly, an ellipsoid-based avoidance method [188] extracts obstacle 3D coordinates from camera images and generates motion vectors for cost-based avoidance path planning.

Vector field methods enhance real-time maneuverability. A depth-camera-based system [168] utilizes OctoMap for perception, computing tangent velocity vectors to guide UAVs through unknown environments. For swarm navigation, a vector field histogram (VFH)-based method [230] integrates Apollonius circles for maintaining safe UAV spacing, improving cooperative flight in complex environments. A fusion of VFH+ and dynamic window approaches (DWA) [231] further refines cooperative multi-UAV obstacle avoidance by optimizing azimuth corrections and rotation cost evaluations.

To address high-speed dynamic obstacle avoidance, a chance-constrained approach [232] predicts obstacle velocities, integrating velocity uncertainty into UAV path planning. These enhancements ensure UAVs can react efficiently to both static and moving obstacles in unpredictable environments.

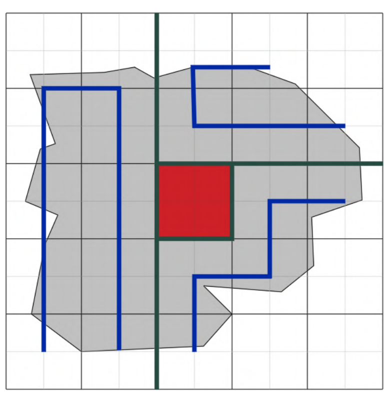
These methods collectively contribute to more robust UAV obstacle avoidance. Table 3.9 compares vector/potential field based techniques in terms of strengths, weaknesses, and applications.

Method	Strengths	Weaknesses	Case Study
Enhanced APF	Flexible, handles formations	Struggles with local minima	[224, 225, 226]
Fuzzy APF	Smooth, adaptive to dynamics	Requires parameter tuning	[227, 228, 229]
Graph-Based	Optimized path planning	Computational overhead	[160, 188]
Vector Field	Real-time, suitable for unknown areas	Dependent on sensor accuracy	[168, 230]
VFH + DWA	Cooperative swarm navigation	Increased processing complexity	[231]
Chance Constraints	Accounts for velocity uncertainty	Requires real-time obstacle tracking	[232]

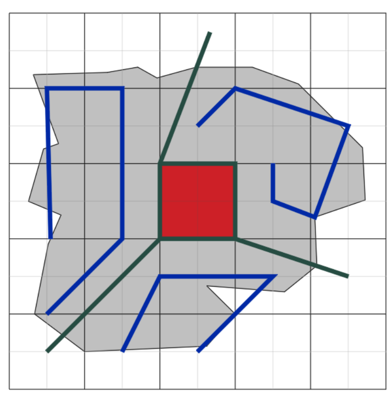
Table 3.9 Comparison of Vector/Potential Field UAV Methods

3.2.5 NON FLYING ZONES

While planning paths for multiple or single UAVs, avoiding NFZs is one of the major challenges. As mentioned previously NFZs are areas that UAVs must not cover or pass over it. NFZs are considered 2D obstacles and are avoided using different methods, such as area partitioning. Figure 3.14 shows an example of area partitioning, where the partitioning borders are shown in dark green color, NFZs are represented by red cells, and the UAV planned paths are shown in blue.



(a) Area partitioning in [42]



(b) Area partitioning in [63, 64]

Figure 3.14 An example on area partitioning and NFZ avoidance.

GRID-BASED AREA PARTITIONING AND PATH PLANNING

A path-planning algorithm for single and multiple UAVs in different area shapes and decomposition (exact and cellular) in the presence of NFZ is proposed in [42]. To avoid NFZ, a new area partitioning method is presented. Using this method, the entire area is divided into equal subareas. Partition borders are set on grid cell borders, to avoid omitting area cells.

Figure 3.14a shows the partitioning area without omitting any grid cells, whereas in Figure 3.14b the partitioning border passes through grid cells which are subsequently omitted or ignored. The coverage quality will decrease when the boundary cells are omitted, as in [63, 64]. In each partitioned area, a grid is formed, and a graph is then generated using the selected vertices. A sub-graph is created by weighing all edges in the graph. The path is optimized by removing insignificant edges and nodes using a filtering technique. The goal is to avoid NFZ's and enable UAVs to plan paths with minimum energy consumption, number of turns, and completion time while increasing the quality of coverage.

An area allocation algorithm based on a clustering method and a graph method is proposed by [151]. The partitioning algorithm allows the UAVs to plan paths that deal with obstacles and NFZs within the area of interest. This algorithm provides collision-free subareas without the need to perform visibility tests. The planned paths within each subarea are evaluated by their energy consumption and time to completion.

In [52], the top view of the area is divided into grids and the paths are planned using the Energy-aware grid-based coverage path planning (EG-CPP) algorithm. This algorithm helps in planning coverage paths over irregularly shaped areas while minimizing energy consumption. This algorithm plans collision-free paths that avoid NFZ and 2D obstacles in an area for a single UAV.

OPTIMIZATION ALGORITHMS

The Parallel Self-Adaptive Ant Colony Optimization Algorithm (PSAACO) for coverage path planning is presented in [233]. PSAACO aims to minimize energy consumption and completion time while considering various factors and employs grid mapping, inversion, and insertion operators, self-adaptive parameter tuning, and parallel computing to achieve this goal. The authors also propose the Dynamic Floyd Algorithm (DFA) to efficiently avoid NFZs in coverage path planning (CPP) problems. They demonstrate the effectiveness of PSAACO and DFA through experiments.

A Simultaneously Transmitting and Reflecting Reconfigurable Intelligent Surface (STAR-RIS) is explored in [234] as a way to enhance UAV-enabled wireless communication networks (WPCNs) to avoid NFZs in an area of interest. Moreover, an alternating optimization (AO) algorithm, combining the penalty method and successive convex approximation (SCA) techniques, is proposed to optimize the UAV trajectory, resource allocation, and beamforming vectors for maximizing system performance.

The study in [235] addresses dynamic factors such as NFZs and reappearing targets, using a preprocessing and group-crossover non-dominated sort genetic algorithm II (PGC-NSGAII) for optimized UAV deployment, demonstrating superior performance.

An Obstacle Avoidance-Enabled Consensus (OAEC) algorithm that integrates NFZ avoidance into UAV path planning in [236]. This algorithm extends the standard consensus approach by adding a mechanism to consider NFZs. When UAVs are near NFZs, temporary target points are generated within safe zones to guide them around the NFZs. This ensures that UAVs maintain a safe distance from restricted areas while achieving their mission objectives.

The multistep particle swarm optimization (MPSO) algorithm is used to determine waypoints that avoid NFZs, ensuring that the UAV formation paths are both efficient and safe.

Approximation algorithm iteratively constructs flying tours for each UAV by solving an orienteering problem, ensuring that the total reward collected is maximized within the UAV's energy capacity in [237]. This involves defining a reward function for each Points of Interest (PoI) and optimizing the UAVs' paths to maximize the sum of these rewards while respecting energy constraints. It focuses on optimizing UAV paths to maximize monitoring rewards while adhering to energy constraints, which involves consideration of obstacle and NFZ avoidance to ensure feasible and efficient paths.

ML AND AI METHODS

A multi-agent reinforcement learning approach, which involves training UAVs to recognize NFZs as areas to avoid during their trajectory optimization process presented in [238]. NFZs are incorporated into the simulated environment as obstacles or areas with negative rewards. The UAVs learn to avoid these zones as part of their path optimization strategy. By continuously interacting with the environment, UAVs adapt their paths dynamically to find the most efficient routes that circumvent NFZs.

Markov Decision Process (MDP) is employed in [239] to model the decision-making process for UAVs, where NFZs are integrated into this model as high-cost areas and effectively guiding UAVs to avoid them. NFZs are associated with high penalty costs in the cost function, encouraging UAVs to choose alternate paths that maintain formation while steering clear of these restricted areas. UAVs use predicted state vectors to assess future positions and avoid NFZs by selecting paths with the lowest associated penalties.

Authors in [240] focus on designing robust trajectories and resource allocation strategies for UAV-enabled wireless networks to maximize spectral efficiency while addressing practical challenges like UAV positioning errors and NFZs. This deep learning framework effectively models UAV trajectory and resource allocation, ensuring reliable NFZ avoidance.

SEARCH AND HEURISTIC ALGORITHMS

To integrate UAV navigation safely and efficiently into smart cities, a modified A* search algorithm is presented in [241] to enhance using the Haversine and Vincenty formulas to avoid NFZs and calculate distances on the Earth's surface, accounting for its elliptical shape.

The RRT algorithm employed in [242] to ensure that waypoints are validated against NFZ boundaries. By casting rays, the algorithm checks for intersections with NFZs and adjusts paths accordingly. Implements probabilistic constraints to handle uncertainties in the environment, ensuring a higher probability that UAVs avoid NFZs (Figure 3.15). This involves adjusting the potential fields based on the likelihood of encountering NFZs.

NOVEL AND HYBRID APPROACHES

Dynamic geo-fences are used in [243] to manage UAV paths in real-time, particularly when unexpected NFZ arise. This system enables UAVs to autonomously adjust their paths in response to new geo-fences. In [244], a zoning system with government and user-controlled authorization zones regulates drone fly-overs, using Attribute-Based Access Control (ABAC) and a variation of the A* algorithm for dynamic path planning.

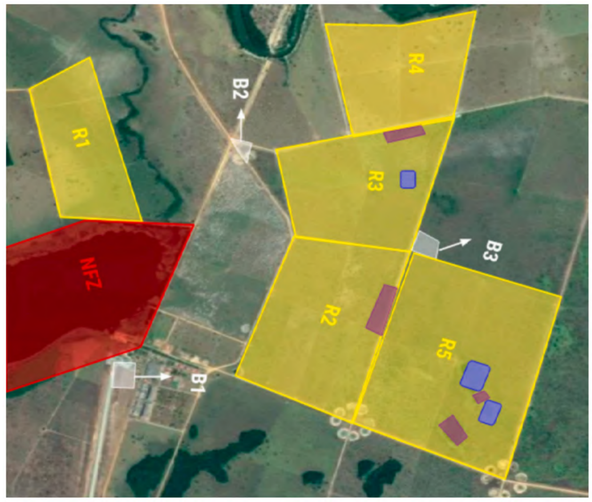


Figure 3.15 Farm scenarios for task planning: key regions, NFZ, and bases from [242].

The study in [245] investigate the design of UAV trajectories and user assignments for downlink data transmission networks while strictly adhering to multiple NFZs. Similarly, a collision avoidance algorithm for UAVs is presented in [246], which ensures minimum separation between UAVs and obstacles, including multiple NFZs, using differential geometry to calculate the necessary heading angle changes. This method efficiently navigates complex environments by avoiding NFZ's. A navigation and collision avoidance framework using an improved Deep Q-Network (DQN) and Faster R-CNN model allows UAVs to autonomously avoid obstacles in [247], demonstrating superior performance in simulated 3D environments.

The challenge that existing mathematical approaches to avoid NFZs throughout their UAVs path is addressed in [248]. The conventional method only ensures that UAVs avoid NFZs at discrete points. To overcome this limitation, the authors introduce a new constraint for expanded NFZs that ensures complete avoidance.

3.2.6 REVIEW AND ANALYSIS

A successful ODA strategy consists of several key stages, as illustrated in Figure 3.16. UAVs are often deployed for missions involving scanning, coverage, or navigation through a designated area. To define the flight path, the mission plan is typically designed using a 2D or 3D map, which provides spatial awareness of the environment.

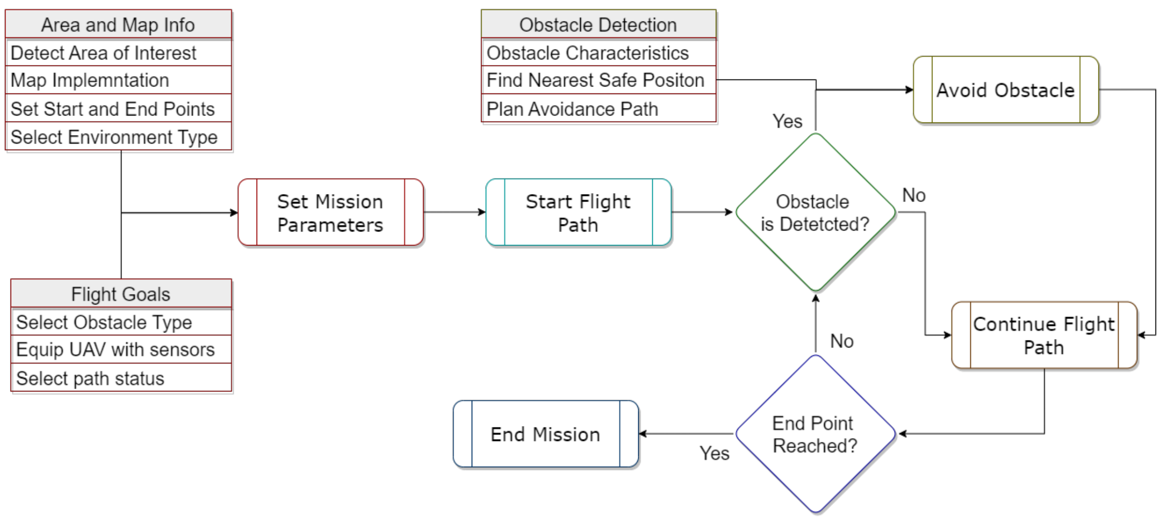


Figure 3.16 ODA Process Steps

UAVs must detect and avoid static and dynamic obstacles. Pre-known obstacles are handled via offline planning, while unexpected ones require real-time avoidance. Upon detection, the UAV adjusts its trajectory and resumes its path. Obstacle detection relies on sensors like IR, LIDAR, Ultrasonic, RGB-D, or stereo cameras, each varying in cost, weight, power, range, and 3D imaging capability [175].

Avoidance is influenced by energy consumption, flight duration, path efficiency, turning angles, and environmental factors like wind. These challenges intensify in complex 3D environments with dynamic obstacles, requiring adaptive, real-time decision-making.

ODA APPROACHES REVIEW AND ANALYSIS

In this section, we compare and analyze the ODA approaches reviewed in this survey based on different criteria. Table 3.10 presents this comparison and includes information on the types of obstacles the approach is designed to detect and avoid (static, dynamic, or both), the environment in which the approach is intended to operate (2D or 3D), the key performance metrics that are considered, whether experiments or testing were conducted in real-life or on a simulator and if paths are planned in offline or online mode. Table 3.11 illustrates a timeline of obstacle detection and avoidance (ODA) methods used in UAVs over the last decade, based on recent advancements.

The present survey reveals that the majority of the works (66%) detected and avoided obstacles in a 3D environment. On the other hand, 28% relied on a 2D environment, while 6% used a combination of 2D and 3D which is more realistic. In addition, there are few energy-aware approaches (7.35%) that require more attention to be paid to this key performance by researchers, as it plays a key role in every mission. The static obstacle type is covered by 29.4% of the approaches, where dynamic obstacles are detected and avoided in 23.5% of the works, however, both dynamic and static obstacles are taken into consideration in 47% of the approaches.

The majority of approaches (78%) equipped UAVs with various types of sensors, while 22% of works did not use sensors or assumed that obstacle data is already known. Maps are used in 71.5% of approaches to detect and avoid obstacles. As for the path planning mode, 47% of the work was done online and 7.35% offline, 13.35% of the work combined online and offline planning, and 32.35% didn't provide any information. Out of all these paths, 21.7% were tested in real life, 50% in simulators, and 28.3% both in real life and simulators.

Table 3.10 Comparison between approaches related to ODA for UAVs

ODA Case Studies References		[53]	[154]	[155]	[157]	[158]	[160]	[161]	[165]	[166]	[167]	[168]	[169]	[170]	[171]
Obstacle Type	Static	✓	✓	✓	✓	-	✓	-	✓	✓	-	✓	✓	-	✓
	Dynamic	-	✓	-	-	✓	✓	✓	-	-	✓	-	-	✓	✓
Environment	2D	✓	-	-	-	-	-	✓	-	✓	-	-	-	-	-
	3D	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	✓
Sensors	Stereo Cam	-	-	✓	-	-	-	-	✓	✓	✓	-	✓	✓	✓
	Monocular Cam	✓	✓	-	-	-	-	-	-	-	-	-	-	-	-
	RGB-D Cam	-	-	-	✓	-	-	-	-	-	-	✓	-	-	-
	FPV Cam	-	-	-	-	✓	-	✓	-	-	-	-	-	-	-
	Ultrasonic	-	-	-	✓	-	-	-	-	-	-	-	-	-	-
	Infrared	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Lidar Laser	-	-	-	-	-	-	-	-	-	-	-	-	✓	✓
	MIMO Radar	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Map Type	Grid-Based	✓	-	-	-	-	-	-	-	-	-	-	-	-	-
	Depth	-	✓	-	-	✓	-	-	-	-	-	-	-	-	-
	Octo	-	-	✓	-	✓	-	-	-	-	✓	✓	-	✓	-
	RTAB	-	-	-	✓	-	-	-	-	-	-	-	-	-	-
	Disparity	-	-	-	-	✓	-	-	✓	-	-	-	✓	-	-
	UV	-	-	-	-	-	-	-	✓	-	-	-	-	-	-
	Other	-	-	-	-	-	✓	✓	-	✓	-	-	-	-	-
Path Status	Online	✓	-	-	✓	✓	✓	-	✓	✓	✓	✓	-	-	✓
	Offline	✓	-	✓	-	-	✓	-	✓	-	-	✓	-	-	✓
	N/A	-	✓	-	-	-	-	✓	-	-	-	-	✓	✓	-
KPIs	Energy	✓	-	-	-	-	-	-	✓	-	-	-	-	-	-
	Time	✓	-	-	-	-	✓	✓	✓	-	✓	✓	-	✓	✓
	Path Length	✓	✓	-	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Turning Angles	✓	-	✓	-	-	-	-	-	-	✓	-	-	-	✓
Evaluation	Real-life	-	✓	✓	-	-	✓	-	✓	-	✓	✓	✓	-	✓
	Simulator	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	-	✓	✓

✓ Check mark symbol indicates the feature presence. - The dash symbol indicates the absence of a feature.

Continue Table 3.10: Comparison between approaches related to ODA for UAVs.

ODA Case Studies References		[172]	[173]	[174]	[175]	[176]	[177]	[178]	[179]	[119]	[180]	[181]	[182]	[183]	[184]
Obstacle Type	Static	✓	✓	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓
	Dynamic	✓	-	-	-	✓	-	✓	-	✓	✓	✓	✓	✓	-
Environment	2D	-	-	-	✓	-	-	-	✓	-	-	-	-	✓	-
	3D	✓	✓	✓	-	✓	✓	✓	-	✓	✓	✓	✓	-	✓
Sensors	Stereo Cam	✓	✓	✓	-	-	-	-	-	-	-	-	-	-	-
	Monocular Cam	-	-	-	✓	✓	✓	✓	✓	✓	-	-	-	-	-
	RGB-D Cam	-	-	-	-	-	-	-	-	-	✓	✓	-	-	-
	FPV Cam	-	-	-	-	-	-	-	-	-	-	-	✓	✓	✓
	Ultrasonic	-	-	✓	-	-	-	-	-	-	-	-	-	-	-
	Infrared	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Lidar Laser	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	MIMO Radar	-	-	-	-	✓	-	-	-	-	-	-	-	✓	-
Map Type	Grid-Based	-	-	-	-	-	-	-	-	✓	-	-	-	-	-
	Depth	✓	✓	✓	-	-	-	-	-	-	-	-	-	-	✓
	Octo	-	-	-	-	✓	-	-	-	-	-	-	-	-	-
	RTAB	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Disparity	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	UV	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Other	-	-	-	✓	-	-	✓	✓	-	-	✓	-	✓	-
Path Status	Online	-	-	✓	-	✓	✓	-	✓	✓	-	✓	✓	✓	-
	Offline	-	-	-	✓	-	-	✓	-	-	-	-	✓	✓	-
	N/A	✓	✓	-	-	-	-	-	-	-	✓	-	-	-	✓
KPIs	Energy	-	-	-	-	-	-	-	-	✓	-	-	-	-	-
	Time	-	-	-	-	-	✓	-	-	-	✓	✓	✓	✓	✓
	Path Length	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
	Turning Angles	✓	-	-	✓	✓	-	-	-	-	✓	-	-	-	✓
Evaluation	Real-life	-	-	✓	✓	✓	✓	-	✓	✓	-	✓	-	-	-
	Simulator	✓	✓	-	-	✓	-	✓	-	-	✓	✓	✓	✓	✓

✓ Check mark symbol indicates the feature presence. - The dash symbol indicates the absence of a feature.

Continue Table 3.10: Comparison between approaches related to ODA for UAVs.

ODA Case Studies References		[185]	[186]	[187]	[188]	[189]	[190]	[191]	[199]	[201]	[200]	[202]	[203]	[204]	[205]
Obstacle Type	Static	✓	✓	-	✓	-	-	✓	✓	✓	✓	✓	-	✓	✓
	Dynamic	-	✓	✓	✓	✓	✓	-	✓	✓	-	✓	✓	-	✓
Environment	2D	-	-	-	-	✓	-	✓	✓	-	-	-	-	✓	✓
	3D	✓	✓	✓	✓	-	✓	-	-	✓	✓	✓	✓	-	-
Sensors	Stereo Cam	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Monocular Cam	-	-	-	✓	-	-	-	✓	-	✓	-	-	-	-
	RGB-D Cam	-	-	-	-	-	-	-	-	✓	-	✓	-	-	-
	FPV Cam	✓	✓	-	-	-	-	-	-	-	-	-	-	-	-
	Ultrasonic	-	-	-	-	✓	✓	✓	-	-	-	-	-	-	-
	Infrared	-	-	-	-	-	✓	✓	-	-	-	-	-	-	-
	Lidar Laser	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	MIMO Radar	-	-	✓	-	-	-	-	-	-	-	-	-	-	-
Map Type	Grid-Based	-	-	-	-	-	-	-	✓	-	-	-	-	-	-
	Depth	✓	-	-	✓	-	-	-	-	✓	✓	-	-	-	-
	Octo	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	RTAB	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	Disparity	-	-	-	-	-	-	-	-	-	-	-	-	-	-
	UV	-	-	-	-	-	-	-	-	✓	-	-	-	-	-
	Other	-	✓	-	-	-	-	✓	-	-	✓	✓	✓	-	-
Path Status	Online	✓	-	-	-	-	-	✓	-	✓	✓	✓	✓	-	-
	Offline	-	-	-	-	-	-	-	✓	-	-	-	-	-	✓
	N/A	-	✓	✓	✓	✓	✓	-	-	-	-	-	-	✓	-
KPIs	Energy	-	-	-	-	-	-	-	-	-	✓	-	-	-	-
	Time	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	-	-	-
	Path Length	-	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	✓
	Turning Angles	-	✓	✓	-	-	✓	✓	✓	-	✓	-	✓	-	-
Evaluation	Real-life	✓	-	✓	-	✓	-	✓	-	✓	✓	-	-	-	-
	Simulator	✓	✓	-	✓	✓	✓	-	✓	✓	-	✓	✓	✓	✓

✓ Check mark symbol indicates the feature presence. - The dash symbol indicates the absence of a feature.

Continue Table 3.10: Comparison between approaches related to ODA for UAVs.

ODA Case Studies References		[249]	[211]	[206]	[207]	[208]	[209]	[210]	[213]	[215]	[218]	[222]	[250]	[251]
Obstacle Type	Static	✓	✓	-	-	-	✓	✓	✓	✓	✓	✓	✓	✓
	Dynamic	✓	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	-
Environment	2D	✓	-	✓	-	-	✓	-	-	✓	✓	-	-	-
	3D	✓	✓	-	✓	✓	-	✓	✓	-	-	✓	✓	✓
Sensors	Stereo Cam	-	-	-	-	-	-	-	✓	-	-	-	-	-
	Monocular Cam	-	-	-	✓	-	✓	-	-	-	✓	✓	✓	-
	RGB-D Cam	-	-	-	-	-	-	✓	-	-	-	-	-	✓
	FPV Cam	-	-	-	-	✓	-	-	-	-	-	-	-	
	Ultrasonic	-	-	-	-	✓	-	-	-	-	-	-	✓	✓
	Infrared	-	-	-	-	-	-	-	-	-	-	-	-	-
	Lidar Laser	-	-	-	-	✓	-	-	-	✓	-	-	-	-
	MIMO Radar	-	-	-	-	-	✓	-	-	-	-	-	-	-
Map Type	Grid-Based	-	-	-	-	-	-	-	-	-	-	-	-	✓
	Depth	-	-	-	-	-	-	✓	✓	-	-	-	-	-
	Octo	-	-	-	-	-	-	-	-	-	-	-	-	-
	RTAB	-	-	-	-	-	-	-	-	-	-	-	-	-
	Disparity	-	-	-	-	-	-	-	-	-	-	-	-	-
	UV	-	-	-	-	-	-	-	-	-	-	-	-	-
	Other	-	✓	✓	✓	✓	✓	✓	-	✓	-	✓	✓	-
Path Status	Online	✓	✓	-	✓	-	-	✓	✓	-	✓	✓	✓	✓
	Offline	✓	✓	-	-	-	-	-	-	-	-	-	-	-
	N/A	-	-	✓	-	✓	✓	-	-	✓	-	-	-	-
KPIs	Energy	-	-	-	-	-	-	-	-	-	-	-	-	-
	Time	-	✓	✓	✓	✓	✓	✓	-	✓	-	✓	-	-
	Path Length	-	-	-	-	✓	-	✓	✓	-	-	-	-	✓
	Turning Angles	✓	-	✓	✓	✓	✓	-	-	✓	-	✓	-	✓
Evaluation	Real-life	-	✓	-	-	-	-	-	✓	✓	✓	✓	✓	-
	Simulator	✓	✓	✓	✓	✓	✓	✓	✓	✓	-	-	✓	✓

✓ Check mark symbol indicates the feature presence. - The dash symbol indicates the absence of a feature.

Continue Table 3.10: Comparison between approaches related to ODA for UAVs.

ODA Case Studies References		[219]	[220]	[221]	[223]	[224]	[225]	[226]	[227]	[228]	[229]	[230]	[232]	[231]
Obstacle Type	Static	-	✓	✓	✓	✓	-	✓	-	✓	✓	✓	-	✓
	Dynamic	✓	✓	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓
Environment	2D	-	✓	-	-	✓	✓	-	-	✓	✓	✓	-	✓
	3D	✓	-	✓	✓	-	-	✓	✓	-	-	-	✓	✓
Sensors	Stereo Cam	-	-	-	-	-	-	-	-	-	-	-	-	-
	Monocular Cam	-	-	-	-	-	-	-	-	-	-	-	-	-
	RGB-D Cam	-	-	-	✓	-	-	-	-	-	-	-	-	-
	FPV Cam	-	-	-	-	-	-	-	-	✓	-	-	✓	-
	Ultrasonic	-	-	-	-	-	-	-	-	-	-	-	-	-
	Infrared	-	-	-	-	-	-	-	-	-	-	-	-	-
	Lidar Laser	-	-	-	-	-	-	-	-	✓	-	✓	-	-
	MIMO Radar	-	-	✓	-	-	-	-	-	-	-	-	✓	-
Map Type	Grid-Based	-	✓	-	-	-	-	-	-	-	-	✓	-	-
	Depth	-	-	-	-	-	-	-	-	-	-	-	-	-
	Octo	-	-	-	-	-	-	-	-	-	-	-	-	✓
	RTAB	-	-	-	-	-	-	-	-	-	-	-	-	-
	Disparity	-	-	-	-	-	-	-	-	-	-	-	-	-
	UV	-	-	-	-	-	-	-	-	-	-	-	-	-
	Other	-	-	-	✓	-	-	-	-	-	-	-	-	-
Path Status	Online	-	✓	✓	-	✓	✓	✓	-	✓	✓	✓	-	✓
	Offline	-	-	-	-	-	-	-	-	-	-	-	-	-
	N/A	✓	-	-	✓	-	-	-	✓	-	-	-	✓	-
KPIs	Energy	-	-	-	-	-	-	-	-	-	✓	-	-	-
	Time	✓	✓	✓	✓	✓	✓	✓	-	-	-	-	✓	✓
	Path Length	-	✓	-	-	-	-	-	✓	-	✓	✓	-	-
	Turning Angles	-	-	✓	-	✓	-	✓	-	-	-	✓	-	✓
Evaluation	Real-life	-	-	-	✓	-	-	✓	-	-	-	-	-	-
	Simulator	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	✓	✓	✓

✓ Check mark symbol indicates the feature presence. - The dash symbol indicates the absence of a feature.

It's important to note that using simulators alone does not guarantee that the methodologies will perform effectively and accurately in real-life scenarios. Out of all the works, 42.65% underwent testing in real-life scenarios or both in simulation and real life. This had an impact on their maturity level, which is determined by the method's success in real life.

From the data presented in Figure 3.17, we see that a significant portion of the works reviewed in this study proposed methods for avoiding obstacles during autonomous navigation. Specifically, 25% of the works rely on machine learning techniques and 19% map data to avoid obstacles.

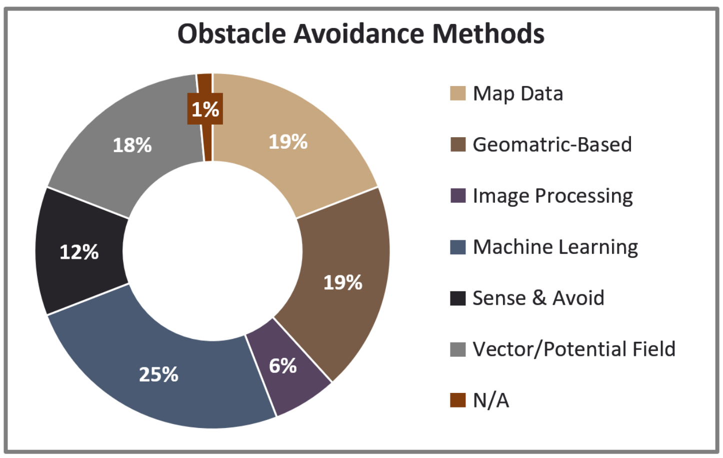


Figure 3.17 Distribution of obstacle avoidance methods.

Furthermore, 19% of the works utilized geometrical-based methods, which may involve the use of geometric shapes or mathematical calculations to determine the location and path of obstacles. In 19% of the works involves using vector or potential fields to guide the movement of the autonomous system around obstacles.

On the other hand, 12% of the approaches employed to sense and avoid methods, in which the autonomous system is equipped with sensors that allow it to detect and avoid obstacles. In addition, 6% of the works used image processing techniques to identify and avoid obstacles, while 1% of the works did not specify any details about the method they used.

Table 3.11 presents a timeline of ODA methods used for UAVs from 2015 to 2024. The ODA methods are classified into categories such as Machine Learning & AI, Sensor Fusion, SLAM, Reactive Control, Bio-Inspired, Collaborative UAV System, Single Sensor Navigation, Energy-Aware, Perception, and Computer Vision & VR.

Notable progress includes the emergence of Machine Learning & AI studies between 2019-2024, Perception algorithms from 2015-2024. Sensor Fusion advancements in 2015, 2017, 2018, 2020, 2021, and 2024, also Reactive Control studies in 2017,2018, and 2020-2024. Moreover, the SLAM based works present in years 2015, 2018, 2019, 2023, and 2024. Singles sensor navigation works appears in years between 2017 - 2019. The table provides a comprehensive overview of the evolution of ODA techniques in UAVs over the recent 10 years.

Table 3.11 Timeline for the evolution of the ODA approaches in the last decade based on the recent advances in UAVs

Years	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024
Machine Learning & AI	-	-	-	-	[154, 184, 219]	[166, 178, 181, 222]	[172, 173, 179]	-	[205, 220]	[250]
Sensor Fusion	[190]	-	[191]	[170, 171, 174]	-	[176]	[208]	-	-	[157]
SLAM	[190]	-	-	[168, 200]	[158]	-	-	-	[202]	[221]
Reactive Control	-	-	[169, 207]	[168, 224]	-	[227]	[225, 70]	[160]	[201, 228, 231, 210]	[199]
Bio-Inspired	-	-	[175]	-	-	-	[167]	-	-	-
Collaborative UAV system	-	-	-	-	-	-	[209]	[161]	-	-
Single Sensor Navigation	-	-	[182]	[185]	[187, 189]	-	-	-	[232]	-
Energy-Aware	-	-	-	[165]	[119]	[53]	-	-	-	-
Perception	[252, 180]	[223]	[169, 204]	[168]	[188]	[161]	[211]	[218]	[251, 230]	[226, 250]
Computer Vision & VR	-	-	[186, 213]	[185]	-	-	-	[218]	-	-

NFZ AVOIDANCE APPROACHES REVIEW AND ANALYSIS

Based on various criteria, we compare and analyze the approaches reviewed in this survey for avoiding NFZs. Table 3.12 compares these approaches based on the type of area decomposition used, the number of NFZs avoided, the method used to partition the area, the number of UAVs involved, and the KPIs used to evaluate the approaches.

Around 28.6% of the works uses approximate cellular decomposition and grid division. Of these approaches, 9.5% use exact and approximate area decomposition with the application of grid sub-division techniques in 4.8% of all works. In 47.6% of the works grid-based technique is applied and 19% used other techniques. However, 66.66% and 23.8% of all the works did not include any information about area cellular decomposition and decomposition technique respectively. Out of all the approaches considered, 76.2% avoid one NFZ within the area, while 28.6% only avoid NFZs around the area. However, 71.4% of the approaches are able to avoid more than one NFZ within the area.

Additionally, 43% of the reviewed works did not provide or use an existing partitioning algorithm for the area of interest to avoid NFZs. Among the works only 19% mentioned that the area partitioning algorithm doesn't exclude cells that lie under the partitioning border.

Only 4.8% of the approaches provided an NFZ avoidance algorithm that could be used for single and multiple numbers of UAVs. Let us note that only 38.1% of the approaches are energy-aware, and 85.7% of the works have considered completion time in their approach. Moreover, 28.6% of the works take the quality of coverage as a KPI. Among 14.3% of the studies none of these three main KPI's is taken into account.

Table 3.12 Comparison between approaches related to NFZ Avoidance for UAVs

NFZ Case Studies References		[42]	[63]	[42]	[151]	[52]	[233]	[234]
Area Cellular decomposition	Approximate	✓	✓	✓	✓	✓	✓	-
	Exact	✓	-	-	-	-	-	-
Area decomposition Technique	Grid-based	✓	✓	✓	✓	✓	✓	-
	Grid Sub-division	✓	-	-	-	-	-	-
	Other	-	-	-	-	-	-	-
Non-flying zones	Presence of NFZ inside the area	✓	✓	✓	✓	✓	✓	✓
	NFZ located around the area	✓	-	-	✓	✓	✓	-
	More than one NFZ inside the area	-	-	-	✓	-	✓	-
Area Partitioning	Provide Partition algorithm	✓	✓	✓	✓	-	✓	-
	Exclude partition border cells	-	✓	✓	-	-	✓	-
Number of UAVs	Single UAV	✓	-	-	-	✓	✓	✓
	Multiple UAVs	✓	✓	✓	✓	-	-	-
KPI	Energy consumption	✓	-	-	✓	✓	✓	✓
	Completion time	✓	✓	✓	✓	✓	✓	✓
	Quality of Coverage	✓	✓	✓	-	-	-	-

✓ Check mark symbol indicates the feature presence. - The dash symbol indicates the absence of a feature.

Table 3.12 Continuing Table 3.12

NFZ Case Studies References		[235]	[236]	[237]	[238]	[239]	[240]	[241]
Area Cellular decomposition	Approximate	-	-	-	-	-	-	-
	Exact	-	-	-	-	-	-	-
Area decomposition Technique	Grid-based	-	-	✓	-	-	-	✓
	Grid Sub-division	-	-	-	-	-	-	-
	Other	✓	✓	-	-	-	✓	-
Non-flying zones	Presence of NFZ inside the area	✓	✓	✓	-	-	✓	✓
	NFZ located around the area	-	-	-	-	-	-	✓
	More than one NFZ inside the area	✓	✓	-	✓	✓	✓	✓
Area Partitioning	Provide Partition algorithm	✓	-	-	-	-	✓	-
	Exclude partition border cells	-	-	-	-	-	-	-
Number of UAVs	Single UAV	-	-	-	-	-	✓	✓
	Multiple UAVs	✓	✓	✓	✓	✓	-	-
KPI	Energy consumption	✓	-	✓	-	-	-	-
	Completion time	✓	✓	✓	✓	-	✓	✓
	Quality of Coverage	✓	-	-	-	-	✓	-

✓ Check mark symbol indicates the feature presence. - The dash symbol indicates the absence of a feature.

Table 3.12 Continuing Table 3.12

NFZ Case Studies References		[242]	[243]	[244]	[245]	[246]	[247]	[248]
Area Cellular decomposition	Approximate	-	-	-	-	-	-	-
	Exact	-	-	-	✓	-	-	-
Area decomposition Technique	Grid-based	✓	-	✓	-	-	-	-
	Grid Sub-division	-	-	-	-	-	-	-
	Other	-	✓	-	✓	✓	-	-
Non-flying zones	Presence of NFZ inside the area	-	✓	✓	✓	✓	✓	✓
	NFZ located around the area	✓	-	-	-	-	-	-
	More than one NFZ inside the area	✓	✓	✓	✓	✓	✓	✓
Area Partitioning	Provide Partition algorithm	✓	✓	✓	✓	-	✓	-
	Exclude partition border cells	✓	-	-	-	-	-	-
Number of UAVs	Single UAV	✓	-	✓	✓	✓	✓	✓
	Multiple UAVs	-	✓	-	-	-	-	-
KPI	Energy consumption	✓	-	-	-	-	-	-
	Completion time	✓	✓	✓	-	✓	✓	-
	Quality of Coverage	-	-	✓	-	-	-	-

✓ Check mark symbol indicates the feature presence. - The dash symbol indicates the absence of a feature.

3.3 SUMMARY

Chapter 3 presents a comprehensive review of the existing literature and technologies related to UAV applications, particularly in healthcare and emergency response. It begins with an exploration of UAVs in healthcare missions, detailing their roles in medical supply delivery, organ transport, blood sample transportation, and COVID-19 response efforts. The chapter also examines the use of UAVs in search and rescue operations, emphasizing their ability to reach inaccessible locations and provide timely assistance.

Furthermore, the chapter discusses various challenges UAVs face in healthcare missions, such as regulatory constraints, technical limitations, and energy efficiency concerns. The review also covers different methodologies in obstacle detection and avoidance (ODA), including machine learning, image processing, and geometric-based approaches. Finally, it evaluates UAV path planning strategies, with a focus on navigation in complex environments and avoidance of no-fly zones (NFZs), highlighting the advantages and limitations of existing solutions.

CHAPTER IV

EA-IPP2N: ENERGY-AWARE INFORMATIVE PATH PLANNING ALGORITHM FOR UAVS

4.1 OVERVIEW

UAVs Applications include agriculture, environmental monitoring, search and rescue (SAR) operations, gas pipeline inspection, and even Mars site exploration [1, 2, 253]. In these operationally critical missions, UAVs are increasingly employed to deliver medical supplies, informative mapping, monitor inaccessible regions, or aid in locating survivors during emergencies.

The effectiveness of such missions depends on the UAV's ability to autonomously plan paths that maximize mission objectives. Autonomous path planning ensures safe and efficient navigation by determining optimal routes while satisfying specific operational constraints such as limited battery capacity, mission duration, and resource usage. This work focuses on *pre-mission* planning, where the UAV's path is computed offline before flight begins, ensuring safe and efficient mission execution under known constraints.

4.1.1 MOTIVATION

A major challenge in UAV deployments is generating paths that maximize information gain or cover high-priority areas while respecting strict energy and time budgets. This challenge is addressed in the field of Informative Path Planning (IPP), where the objective is to guide a UAV to maximize collected information or utility. IPP often maps to the Orienteering

Problem (OP), where the goal is to visit a subset of locations to maximize cumulative reward within a budget typically flight time.

Although several IPP methods exist for UAVs [254, 255, 256, 257, 8, 258], most simplify the problem by considering only one constraint, often time while neglecting energy consumption. Approaches that include energy [258, 259] often ignore maneuvering costs from heading changes or turning angles, leading to unrealistic estimates for rotary-wing UAVs where turns significantly impact energy use [41]. Furthermore, integrating turning dynamics into both time and energy models remains rare, despite its importance for missions like SAR and healthcare logistics [1].

Studies such as Michel *et al.* [260] show that minimizing distance does not necessarily minimize energy consumption, with differences exceeding 14% in 3D environments. In multi-UAV operations, rapid coverage and reduced turning are equally critical for energy and time efficiency [9, 261]. These gaps highlight the need for planning algorithms that jointly optimize time and energy under realistic UAV motion dynamics.

4.1.2 MAIN CONTRIBUTIONS

To address these challenges, we propose **EA-IPP2n**, an Energy-Aware Informative Path Planning algorithm that:

- Simultaneously enforces time and energy constraints for UAV deployment in critical missions.
- Uses a multi-factor edge cost model combining distance and turning angle to jointly evaluate energy and time cost while incorporating node rewards.

- Implements a two-hop look-ahead planning strategy that balances reward maximization with normalized resource usage through an iterative, greedy decision-making process.

EA-IPP2n represents the environment as a weighted graph with nodes as locations of interest and edges encoding transition costs based on both distance and turning angles. Time and energy are normalized using empirically calibrated UAV motion models [262, 263], ensuring operational feasibility. The algorithm is validated through simulations replicating SAR-like environments with skewed reward distributions, and benchmarked against six state-of-the-art methods. Results show EA-IPP2n achieves higher rewards, respects both budgets, and improves performance across multiple metrics.

4.2 BACKGROUND

IPP has been increasingly studied for UAV missions involving terrain monitoring, environmental surveillance, and SAR. Prior research encompasses sampling-based, heuristic-based, and learning-based methods aimed at maximizing information gain under operational constraints such as time or energy.

Sampling-based approaches such as Informed-RRT* have been adapted for dynamic environments [254], while Dubins path-based orienteering models with neighborhood constraints have been employed to generate cost-efficient trajectories under motion limitations [6]. Probabilistic Road-maps (PRM) and Rapidly-exploring Random Trees (RRT) have also been integrated with reactive methods like the Dynamic Window Approach (DWA) to enable real-time obstacle avoidance in cluttered scenarios [254, 264].

Metaheuristic and evolutionary planners, including GA, ACO, and MCTS, have been widely applied for path planing due to their adaptability and scalability.

The GA is a population-based metaheuristic inspired by the principle of natural selection [265]. It evolves a population of candidate solutions (paths) over generations through crossover, mutation, and selection. In UAV path planning, GA can discover effective visitation sequences that maximize collected rewards under constraints.

The Greedy Search strategy offers a simple and fast alternative by making locally optimal decisions at each step, often based on the nearest unvisited neighbor with the highest reward-to-cost ratio [266]. Despite its low computational overhead, Greedy Search is shortsighted and prone to suboptimal global solutions, especially in environments with sparse or clustered reward distributions.

In contrast, ACO simulates pheromone-guided foraging behavior of ants to probabilistically explore paths [267, 268], while MCTS performs roll out simulations to identify high-reward trajectories through statistical evaluation [269]. These methods balance exploration and exploitation more effectively but may still overlook energy dynamics unless explicitly modeled.

Recent efforts have explored deep learning methods, particularly Deep Reinforcement Learning (DRL), for adaptive path planning. Rückin et al [256] combine DRL with dynamic graphs or spatial attention mechanisms to optimize reward acquisition in complex environments. Semantic understanding has also been incorporated through hierarchical and multi-resolution semantic mapping frameworks [257, 8, 50], improving the UAV's ability to prioritize meaningful regions.

Orisatoki [258] introduced the Information-theoretic HIPP algorithm in unknown environments using occupancy grid maps and LiDAR-based exploration. While effective for maximizing mapping coverage, HIPP does not account for UAV constraints such as energy usage or turning costs, making it less suited for time and energy critical missions.

Multi-UAV systems have gained attention for cooperative exploration and information gathering, utilizing decentralized or leader-follower architectures [48, 7]. The Multiple-UAV IPP and Mapping method (MIPP) proposed by Cho *et al.* [48] formulates the mission as a sequence of subproblems that include task location optimization based on mutual information, wind-aware path planning with energy-based cost functions, and Gaussian process mapping for environmental estimation. The approach models routing as a min–max multiple-depot multiple traveling salesman problem, thereby inherently considering mission time constraints for each UAV. However, MIPP evaluates energy primarily as a function of path length and wind alignment, without explicitly modeling maneuvering costs such as turning angles, which are critical in rotary-wing UAV missions. Online IPP frameworks and Bayesian optimization-based planners [270, 271] further improve adaptability by allowing UAVs to update plans based on evolving observations or model uncertainty.

Despite these advancements, existing IPP methods rely on idealized energy consumption models and typically address only a single constraint. Works such as [259, 272] introduce energy-aware formulations but do not account for factors like turning energy a key factor for UAVs. Furthermore, integrated models that simultaneously handle time, energy, and spatial reward optimization remain limited in the literature.

The proposed EA-IPP2n differs from existing methods by explicitly incorporating both time and energy constraints, accounting for turning effort, and using a two-hop look-ahead strategy within a reward-driven planning framework.

4.3 ORIENTEERING PROBLEM

4.3.1 PROBLEM OVERVIEW

In this work, we propose a *global pre-mission informative path planner* that operates on a subset of strategically selected nodes, each representing a location of interest with an predefined reward. The objective is to maximize the cumulative reward collected under strict **time** and **energy** constraints. The UAV is not required to visit all nodes; instead, the planner determines a *feasible, high-utility visitation sequence* that reflects mission priorities.

We extend the classical Orienteering Problem (OP) by incorporating **dual constraints**: completion time and energy consumption. Both are critical in UAV operations, where motion dynamics and payload mass strongly affect endurance. While increasing battery capacity may extend flight time, it also increases overall energy demand due to added weight. This trade-off makes it essential to design routes that maximize information gain within limited resources.

Fig. 4.1 shows an example of the initial mission context, where a heat map illustrates the spatial distribution of reward values across the operational area before discretization into nodes. The continuous spatial priority map is then discretized into N nodes, each assigned a reward according to its corresponding location in the heat map.

Such a representation can encode information provided by first responders in SAR missions, identify points of interest for gas pipeline inspections, highlight critical zones in wildfire monitoring, or prioritize sectors during large-scale search operations.

4.3.2 GRAPH-BASED PROBLEM DEFINITION

In this work, the set of mission nodes and their rewards are predefined and provided directly to the pre-mission planner as a graph. This graph-based representation allows us to formally define the problem as follows. Let the given budgets be the maximum energy consumption, denoted E_{bud}, and the maximum completion time, denoted T_{bud}. The energy consumption depends on two key factors: the distance traveled between two points (d) and the turning angles (θ), associated with energy costs per meter (λ) and per degree of turning (γ). Similarly, the completion time depends on the UAV's speed (v), the total travel distance, the turning angles, and the rotational rate (ω). We model the environment as a graph $G(N, \mathscr{E}, R, C)$, where:

- $N = \{n_1, n_2, \dots\}$ is the set of nodes,
- $\mathscr{E} = \{e_1, e_2, \dots\} \subseteq N \times N$ is the set of undirected edges,
- R denotes the set of rewards assigned to nodes,
- C represents color-coded reward categories, assigned to values ranging from 0.0 to 1.0. These values indicate the initial relative importance of different zones, from purple (very low priority) to red (very high priority), as shown in Fig. 4.1.

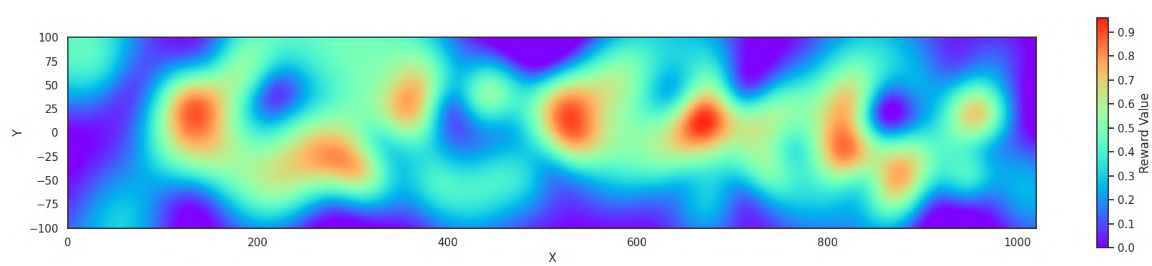


Figure 4.1 Initial mission data visualized as a spatial heatmap, with warmer colors indicating higher-priority areas. (Corresponds to Scenario 1 in Sec. 4.5.)

Each edge is labeled by a mapping $\kappa : \mathscr{E} \rightarrow T \times E$, where $T, E \in \mathbb{R}$ represent the time and energy required to traverse the edge. For example, edge e is labeled as $e(T_e, E_e)$ where

T_e and E_e denote the time and energy required to traverse edge e. Each node is labeled via a mapping $r : N \to R$, indicating the reward value at that location. A path p is as a sequence of k connected edges:

$$p = \{e_1, e_2, \ldots, e_k\} \tag{4.1}$$

The cumulative reward R_p along the path p is the sum of the rewards at each node n_j visited on the path, as follows:

$$R_p = \sum_{n_j \in p} r(n_j) \tag{4.2}$$

where $r(n_j)$ is the reward value associated with node n_j. Our objective is to find the optimal path p^* from the set of feasible paths $\mathscr{P}$ that maximizes the total reward:

$$\begin{aligned} R_{\text{total}} &= \max_{p \in \mathscr{P}} \left\{ \sum_{n_j \in p} r(n_j) \right\} \\ \text{subject to} \quad & \sum_{e \in p} T_e \leq T_{\text{bud}}, \quad \sum_{e \in p} E_e \leq E_{\text{bud}} \end{aligned} \tag{4.3}$$

where T_{bud} and E_{bud} represent the available time and energy budgets, respectively.

4.4 ENERGY-AWARE INFORMATIVE PATH PLANNING USING 2-HOP EVALUATION

The proposed algorithm, EA-IPP2n is a reward-driven path planning algorithm designed for UAV missions operating under strict time and energy constraints. It plans routes through spatial nodes prioritized based on critical area data to maximize total reward while adhering to resource budgets. EA-IPP2n evaluates the nearest 2-hop combinations during each iteration to greedily select the most efficient reward-optimal transitions.

4.4.1 EDGE CALCULATIONS:

Each edge in the graph corresponds to a UAV movement from node j to node k, potentially influenced by the previous heading from node i (i.e., movement from $i \rightarrow j \rightarrow k$). The Euclidean distance between two nodes i and j iwas calculated using Eq(2.3).

The turning angle at node j between nodes $i \rightarrow j \rightarrow k$ is calculated using the cosine law Eq(2.6). If i is `None` (i.e., first step), a virtual node is introduced behind Node 0 to define an initial heading, effectively assigning a default direction (e.g., $(x_i, y_i) = (-1, 0)$) as shown in Fig. 4.2. The result is then converted to degrees.

We use time [263] and energy [262, 41, 53, 42] consumption models to evaluate the transition cost when moving from node j to node k, considering the previous node i. Eq(2.7) and Eq(2.4)

4.4.2 NORMALIZATION OF TIME AND ENERGY

To balance the cost function and make units comparable, time and energy are normalized per edge.

$$\text{MaxTimePerEdge} = \frac{T_{\text{bud}}}{(N-1)/2} \tag{4.4}$$

$$\text{MaxEnergyPerEdge} = \frac{E_{\text{bud}}}{(N-1)/1.5} \tag{4.5}$$

Let T_{e_1} and E_{e_1} be the time and energy required to travel from the current node i to an intermediate node j, and T_{e_2} and E_{e_2} those needed to reach the next candidate node k along the path $i \xrightarrow{T_{e_1},E_{e_1}} j \xrightarrow{T_{e_2},E_{e_2}} k$.

$$NT = \frac{T_{e_1} + T_{e_2}}{2 \cdot \text{MaxTimePerEdge}} \tag{4.6}$$

$$NE = \frac{E_{e_1} + E_{e_2}}{2 \cdot \text{MaxEnergyPerEdge}} \tag{4.7}$$

In the implemented models, both time and energy costs are influenced by path length and turning angles; however, energy consumption is more significantly affected by turns due to the higher power demands of directional changes in rotary-wing UAVs [270, 262]. Observations from the evaluated scenarios (see Sec. 4.5) show that the UAV typically visits only around $N/1.5$ nodes before depleting the energy budget, whereas the time budget often allows for longer trajectories. To reflect this disparity and avoid overly conservative per-edge normalization, we introduce a correction factor of 1.5 in Equation (4.5). Although heuristic, this factor improves the granularity of edge evaluation and is consistent with energy-aware planning strategies that account for realistic UAV motion dynamics [270, 272].

4.4.3 NEIGHBOR FILTERING AND COST FUNCTION SELECTION STRATEGY

To reduce complexity while preserving path quality, the algorithm evaluates only the five nearest neighbors at each hop. This localized neighbor filtering ensures the UAV considers only relevant transitions that are more likely to be feasible given the time and energy constraints. The five-neighbor limit was empirically selected as a trade-off between computational efficiency and solution diversity, particularly in dense graphs where exhaustive edge evaluations would be computationally prohibitive.

To accommodate various trade-offs between operational constraints and mission objectives, the algorithm employs a parametric cost function for evaluating 2-hop transitions (j,k). Each cost function is formulated as a convex combination of normalized time (NT), normalized energy (NE), and average reward $\bar{R}$, as follows:

$$\text{Cost}(j,k) = \alpha \cdot NT + \alpha \cdot NE + \beta \cdot (1 - \bar{R}) \tag{4.8}$$

Here, α equally weights time and energy, while $\beta = 1 - 2\alpha$ assigns the rest to reward, enabling flexible trade-offs between constraints and information gain. The three configurations are shown in Table 4.1.

Table 4.1 Cost function configurations in EA-IPP2n

Cost Function	α	β	**Focus**
$Cost_B$	$\frac{1}{3}$	$\frac{1}{3}$	Balanced trade-off
$Cost_R$	0.20	0.60	Reward-focused
$Cost_C$	0.30	0.40	Constraint-focused

At each step, the algorithm evaluates the best path under each cost configuration and selects the one yielding the highest cumulative reward within the global time and energy budgets. This adaptive multi-cost evaluation improves robustness to varying spatial reward distributions.

The evaluation of candidate two-hop transitions from Node 0 in Fig. 4.2 is presented in Table 4.2. Here, the normalized time (NT) and normalized energy (NE) are computed using Equations 4.6 and 4.7. The average reward ($\bar{R}$) corresponds to the mean of the real reward values assigned to the involved nodes. The cost metrics for each configuration are then obtained using Equation 4.8.

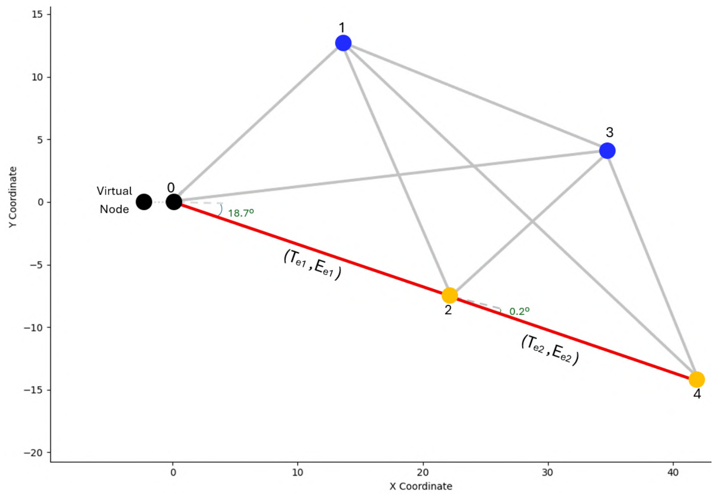


Figure 4.2 Candidate paths from Node 0 to nearby nodes (1–4). The selected path **0 → 2 → 4**, with the lowest cost, is shown in red.

Table 4.2 Evaluation of Paths from Node 0 Using Normalized Metrics and Parametric Cost Functions

$i \to j \to k$	**NT**	**NE**	$\bar{R}$	Cost_B	Cost_R	Cost_C
$0 \to 1 \to 2$	0.62	0.54	0.45	0.713	0.587	0.652
$0 \to 1 \to 3$	0.60	0.46	0.25	0.810	0.727	0.764
$0 \to 1 \to 4$	0.85	0.74	0.50	0.789	0.657	0.700
$0 \to 2 \to 1$	0.64	0.57	0.45	0.744	0.616	0.669
$0 \to 2 \to 3$	0.52	0.42	0.40	0.697	0.635	0.673
$0 \to 2 \to 4$	0.54	0.37	**0.65**	**0.533**	**0.430**	**0.472**
$0 \to 3 \to 4$	0.76	0.63	0.45	0.769	0.618	0.661

After the main loop terminates— once no valid 2-hop pairs remain within budget— the planner attempts to add a final single-hop transition if feasible. This step ensures that any remaining time or energy budget, which may be insufficient for a two-hop sequence but adequate for reaching one more node, is still exploited. By applying the same parametric cost function (adapted to a single hop), the planner can collect additional reward rather than leaving residual resources unused. The proposed planning method is summarized in Algorithm 4.1.

Input: Node set N, depot n_0, budgets $(T_{\text{bud}}, E_{\text{bud}})$, UAV dynamics (v, λ, γ)
Output: Feasible path p^* maximizing cumulative reward

Initialize $p \leftarrow [n_0]$, $R, T, E \leftarrow 0$; set $n_{\text{prev}} \leftarrow$ None, $n_{\text{cur}} \leftarrow n_0$;
Compute MaxTimePerEdge and MaxEnergyPerEdge via Eq(4.4)–Eq(4.5);
while *feasible two-hop* $(i \rightarrow j \rightarrow k)$ *exists* **do**
  Select up to five nearest unvisited neighbors J of n_{cur};
  foreach $j \in J$ **do**
    Select up to five unvisited neighbors K of j;
    foreach $k \in K$ **do**
      Compute (T_{e_1}, E_{e_1}) and (T_{e_2}, E_{e_2}) via Eq(2.4), Eq(2.7);
      if $T + T_{e_1} + T_{e_2} > T_{bud}$ *or* $E + E_{e_1} + E_{e_2} > E_{bud}$ **then**
        discard;
      end
      Compute NT, NE (Eq(4.6)–Eq(4.7)) and $\bar{R} \leftarrow (r(j) + r(k))/2$;
      Evaluate cost using Eq(4.8) for each configuration in Table 4.1;
      Record the lowest-cost (j, k) per configuration;
    end
  end
  Choose the configuration yielding the highest reward among its best (j, k) pairs;
  Append (j^*, k^*) to p, update T, E, R, and mark nodes visited;
  Set $n_{\text{prev}} \leftarrow j^*$, $n_{\text{cur}} \leftarrow k^*$;
end
if *residual budget allows single-hop* **then**
  Evaluate remaining candidates using Eq(2.4)–Eq(4.8); append best k^* to p;
end
return p^*

Algorithm 4.1: EA-IPP2n Planner

4.5 EVALUATION AND RESULTS

4.5.1 EXPERIMENTAL SETUP

The proposed algorithm is benchmarked against HIPP [258], MIPP [48], and four well-known path planning/metaheuristic strategies ACO, GA, Greedy, and MCTS. All algorithms were evaluated on two scenarios with 100 nodes having the same time budget constraints in 5 missions (10 in total). Specifically, if visiting the next candidate node would cause the total path time to exceed the time budget, the algorithm terminates the path. This stopping condition was uniformly applied to ACO, GA, MCTS, Greedy Search, HIPP, and MIPP (which inherently considers time budget constraints).

In addition, we enforced a path length and turning-angle based energy model 2.7 on both HIPP and MIPP to enable a fairer comparison under similar operational conditions. While MIPP originally accounts for path length and wind-field effects in its energy modeling, it does not explicitly capture turning effort. By extending both HIPP and MIPP with the same dual-budget constraint (time and energy, including turning costs), we ensured consistency in performance evaluation across algorithms. Although EA-IPP2n was originally designed to integrate these factors, applying the same enriched formulation to baseline methods highlights the benefits of explicitly accounting for turning-induced energy consumption alongside time constraints. All simulations were conducted on Ubuntu 20.04, running on an Intel i7-6700HQ CPU, 16 GB RAM, and an NVIDIA GTX 965M GPU. The simulation parameters used for each method are as follows

- **Ant Colony Optimization (ACO):** Configured with 50 ants and 100 iterations.
- **Genetic Algorithm (GA):** Utilized a population size of 100 over 200 generations.
- **Monte Carlo Tree Search (MCTS):** Executed 100,000 Monte Carlo simulations per decision step.

4.5.2 TIME AND ENERGY BUDGET SELECTION

To examine the scalability and robustness of the proposed algorithm under varying operational constraints, two scenario sets, denoted SC1 and SC2, were constructed. Each scenario includes the same five missions, with time and energy budgets scaled proportionally to simulate varying resource conditions:

- Mission 1: $T_{\text{bud}} = 900\,\text{s}, \quad E_{\text{bud}} = 216\,\text{kJ}$
- Mission 2: $T_{\text{bud}} = 1050\,\text{s}, \quad E_{\text{bud}} = 252\,\text{kJ}$
- Mission 3: $T_{\text{bud}} = 1200\,\text{s}, \quad E_{\text{bud}} = 288\,\text{kJ}$
- Mission 4: $T_{\text{bud}} = 1350\,\text{s}, \quad E_{\text{bud}} = 324\,\text{kJ}$
- Mission 5: $T_{\text{bud}} = 1500\,\text{s}, \quad E_{\text{bud}} = 360\,\text{kJ}$

The mission budgets are scaled proportionally with respect to Mission 5, using ratios of 0.6, 0.7, 0.8, 0.9, and 1.0 for Missions 1–5, respectively. This scaling ensures that the time budgets span approximately 15 to 25 minutes, aligning with realistic operational constraints for commercial rotary-wing UAVs in SAR or mapping scenarios.

For instance, while the DJI Matrice 300 RTK UAV has a manufacturer-rated maximum flight time of up to 55 minutes under ideal conditions [273], actual endurance with mission-specific payloads and dynamic flight maneuvers is typically limited to 15–25 minutes. Consequently, Mission 5's 1500 s (25 minutes) time budget and 360 kJ energy budget serve as practical upper bounds, with the remaining missions scaled proportionally.

E_{bud} is calculated based on physical battery specifications and validated empirical UAV energy consumption models. For example, a typical 6-cell lithium-polymer (Li-Po) battery with 4500 mAh capacity at 22.2 V nominal voltage stores:

$$E_{\text{total}} = 22.2\,\text{V} \times 4.5\,\text{Ah} \times 3600\,\text{s/h} = 359\,\text{kJ}. \tag{4.9}$$

It is important to note that all missions were simulated within a fixed area of 1000×200 meters with a total of 100 nodes distributed across the environment. As introduced earlier in Section 4.3, the heatmap for SC 1 is shown in Fig. 4.1. Fig. 4.3 presents the corresponding heatmap for SC2, illustrating the distribution of reward values used to generate node locations. The two scenarios differ in node distribution and reward assignments, providing diverse, fair testing conditions to assess robustness across environments.

All algorithms used identical UAV motion parameters for fair comparison. Forward speed was fixed at $v = 2.0\,\text{m/s}$, typical for monitoring or SAR operations requiring stability and maneuverability. The rotation rate was set to $\omega = 60^{\circ}/\text{s}$. Energy consumption was modeled using empirically validated coefficients: linear travel consumes $\lambda = 0.1164\,\text{kJ/m}$, and turning effort consumes $\gamma = 0.0173\,\text{kJ/degree}$ [262].

4.5.3 RESULTS AND PERFORMANCE ANALYSIS

Tables 4.3 - 4.7 presents a comprehensive summary of the simulation outcomes of the two scenarios across all missions. Examples of planned paths for two different missions, SC 1 Missions 2 and 5 and SC 2 Mission 3 represented in Fig. 4.4a, 4.4b, 4.4c, and 4.4d respectively. Across the five missions in both scenarios, EA-IPP2n consistently respected time and energy budgets while maintaining the highest total reward collection.

In SC1, EA-IPP2n achieved the highest total rewards in four out of five missions. In Mission 3, Greedy obtained a slightly higher reward (21.5 vs. 20.4), but it exceeded the energy budget by 69.7% and incurred 36.5% more turning effort than EA-IPP2n, reducing its suitability for budget-constrained missions.

Table 4.3 Performance comparison of algorithms for SC 1 and 2 with Mission 1. $T_{\text{bud}} = 900$ s, $E_{\text{bud}} = 216$ KJ. **Bold**: best, underline: second best, *: exceeded E_{bud}.

Set of Nodes	**SC 1**							**SC 2**						
Algorithm	ACO	GA	Greedy	MCTS	HIPP	MIPP	**Ours**	ACO	GA	Greedy	MCTS	HIPP	MIPP	**Ours**
Path Length (m)	1339.47	1729.97	1694.60	1729.29	1548.19	1538.72	1599.52	1219.35	1727.98	1694.60	1729.29	1578.87	1538.72	1540.74
Turning Angles (°)	1966.51	2056.93	2553.92	1217.45	1993.16	2052.05	1585.27	2072.14	2087.49	2553.92	1217.45	1753.41	2052.05	2088.48
Completion Time (s)	702.51	899.27	889.86	884.93	807.31	803.56	826.18	644.21	898.78	889.86	884.93	818.66	803.56	805.18
Energy (KJ)	189.94	236.95*	241.43*	222.35*	214.69	214.61	213.61	177.78	237.25*	241.43*	222.35*	214.11	214.61	215.47
Nodes Visited	30	31	38	13	25	33	38	29	38	38	13	34	33	44
Total Rewards	11.10	15.50	14.40	5.70	12.00	12.90	**16.90**	10.90	17.70	15.20	5.80	17.40	12.50	**21.40**
Process Time (s)	20.1592	2.6229	**0.0015**	5.0082	0.0164	0.2384	0.0089	22.012	2.550	**0.0013**	4.703	0.0362	0.1231	0.0102

Table 4.4 Performance comparison of algorithms for SC 1 and 2 with Mission 2. $T_{\text{bud}} = 1050$ s, $E_{\text{bud}} = 252$ KJ. **Bold**: best, underline: second best, *: exceeded E_{bud}.

Set of Nodes	**SC 1**							**SC 2**						
Algorithm	ACO	GA	Greedy	MCTS	HIPP	MIPP	**Ours**	ACO	GA	Greedy	MCTS	HIPP	MIPP	**Ours**
Path Length (m)	1400.72	2026.98	1956.15	1979.76	1815.71	1764.29	1887.30	1388.18	2012.38	1956.15	1729.29	1819.21	1764.29	1789.48
Turning Angles (°)	1716.42	2159.51	3165.82	1197.71	2267.24	2579.08	1856.35	1867.21	2589.21	3165.82	1217.45	2229.09	2579.08	2459.26
Completion Time (s)	728.97	1049.48	1030.84	1009.84	945.64	925.13	974.59	725.21	1049.34	1030.84	884.93	946.75	925.13	935.73
Energy (KJ)	192.74	273.30*	282.46*	251.16	250.57	249.98	251.80	193.89	279.03*	282.46*	222.35	250.32	249.98	250.84
Nodes Visited	30	38	45	12	32	40	45	29	37	45	13	39	40	46
Total Rewards	11.10	20.60	17.70	6.60	15.10	15.30	**21.10**	10.70	18.80	18.00	5.80	19.90	16.50	**23.40**
Process Time (s)	22.6274	2.6398	**0.0015**	5.0114	0.0219	0.2048	0.0101	22.0124	2.7008	**0.0016**	4.7220	0.0362	0.1647	0.0103

Table 4.5 Performance comparison of algorithms for SC 1 and 2 with Mission 3. $T_{\text{bud}} = 1200$ s, $E_{\text{bud}} = 288$ KJ. **Bold**: best, underline: second best, *: exceeded E_{bud}.

Set of Nodes	**SC 1**							**SC 2**						
Algorithm	ACO	GA	Greedy	MCTS	HIPP	MIPP	**Ours**	ACO	GA	Greedy	MCTS	HIPP	MIPP	**Ours**
Path Length (m)	1812.32	2336.11	2264.38	2230.42	2051.23	1995.14	2141.32	1465.55	2306.90	2264.38	2249.03	2064.43	1995.14	2113.67
Turning Angles (°)	2961.06	1911.72	3708.96	1287.19	2732.84	3182.16	2185.93	2061.26	2768.06	3708.96	1353.48	2503.95	3182.16	2405.35
Completion Time (s)	955.51	1199.92	1194.01	1136.66	1071.16	1050.61	1107.09	767.13	1199.58	1194.01	1147.07	1073.95	1050.61	1096.93
Energy (KJ)	262.18	305.00*	327.74*	281.89	286.04	287.29	287.07	206.25	316.41*	327.74*	285.20	283.62	287.29	287.64
Nodes Visited	44	35	55	14	37	46	46	31	40	55	14	45	46	52
Total Rewards	16.60	18.30	**21.50**	6.90	17.90	18.60	20.40	11.40	21.60	22.40	7.20	24.10	18.80	**26.10**
Process Time (s)	22.7419	2.8962	**0.0018**	5.9659	0.0220	0.2286	0.0102	24.5504	3.0151	**0.0018**	5.5705	0.0283	0.1868	0.0135

Table 4.6 Performance comparison of algorithms for SC 1 and 2 with Mission 4. $T_{\text{bud}} = 1350$ s, $E_{\text{bud}} = 324$ KJ. **Bold**: best, underline: second best, *: exceeded E_{bud}.

Set of Nodes	**SC 1**							**SC 2**						
Algorithm	ACO	GA	Greedy	MCTS	HIPP	MIPP	**Ours**	ACO	GA	Greedy	MCTS	HIPP	MIPP	**Ours**
Path Length (m)	1816.59	2609.19	2393.04	2650.55	2293.30	2243.34	2389.26	1886.01	2560.79	2393.04	2652.07	2299.83	2243.34	2370.44
Turning Angles (°)	2797.38	2722.40	3866.35	1307.43	3073.28	3627.50	2498.10	2884.00	3781.29	3866.35	1307.15	3129.72	3627.50	2701.79
Completion Time (s)	954.92	1349.97	1260.96	1347.06	1197.87	1182.13	1236.26	991.07	1343.42	1260.96	1347.82	1202.08	1182.13	1230.25
Energy (KJ)	259.85	350.81*	345.44*	331.14*	320.11	323.88	321.33	269.42	363.49*	345.44*	331.31*	321.84	323.88	322.66
Nodes Visited	42	42	57	16	42	54	52	43	49	57	16	52	54	59
Total Rewards	16.00	21.40	22.40	7.80	20.50	21.50	**24.00**	17.20	25.20	23.10	8.00	27.10	22.20	**27.50**
Process Time (s)	25.6830	3.0491	**0.0017**	6.7327	0.0289	0.3825	0.0112	24.550	3.3421	**0.0018**	6.4546	0.0286	0.2149	0.0120

Table 4.7 Performance comparison of algorithms for SC 1 and 2 with Mission 5. $T_{\text{bud}} = 1500$ s, $E_{\text{bud}} = 360$ KJ. **Bold**: best, <u>underline</u>: second best, *: exceeded E_{bud}.

Set of Nodes	SC 1							SC 2						
Algorithm	ACO	GA	Greedy	MCTS	HIPP	MIPP	**Ours**	ACO	GA	Greedy	MCTS	HIPP	MIPP	**Ours**
Path Length (m)	2213.48	2903.27	2841.51	2866.96	2542.40	2503.93	2622.86	2145.53	2870.28	2841.51	2953.76	2552.55	2503.93	2602.54
Turning Angles (°)	3796.19	2874.26	4198.88	1600.06	3529.67	3775.26	2874.63	3600.49	3549.70	4198.88	1204.51	3387.55	3775.26	3185.01
Completion Time (s)	1170.01	1499.54	1490.74	1460.15	1330.03	1314.89	1359.34	1132.77	1494.30	1490.74	1496.95	1332.73	1314.89	1354.35
Energy (KJ)	323.32	387.66*	403.39*	361.40*	357.00	356.77	355.03	312.03	395.51*	403.39*	364.66*	355.72	356.77	358.04
Nodes Visited	57	53	64	17	49	57	64	54	54	64	15	56	57	61
Total Rewards	22.40	<u>26.60</u>	24.40	9.00	24.80	22.30	**30.10**	21.70	27.70	25.40	8.20	<u>28.40</u>	23.20	**28.60**
Process Time (s)	27.0884	3.4684	**0.0020**	7.4058	0.0309	0.2439	<u>0.0130</u>	24.78	3.5796	**0.0020**	6.1844	0.0362	0.2354	<u>0.0122</u>

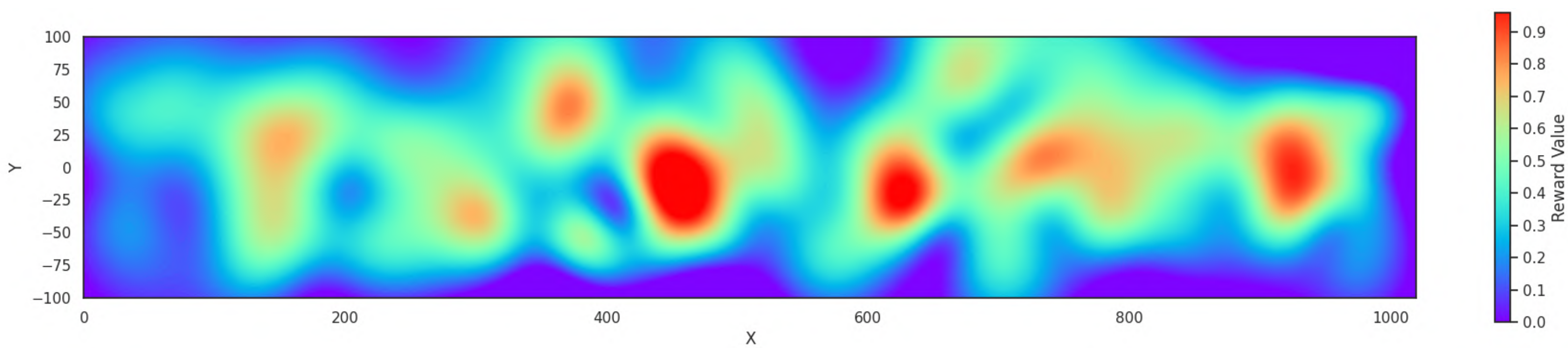
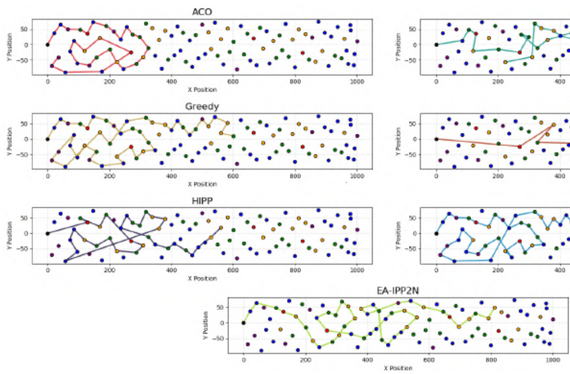
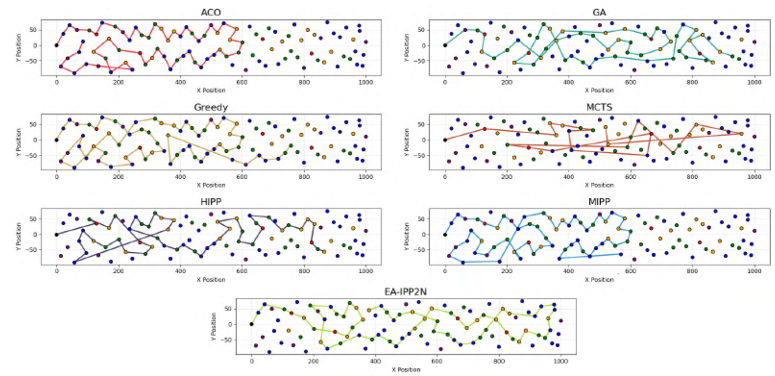


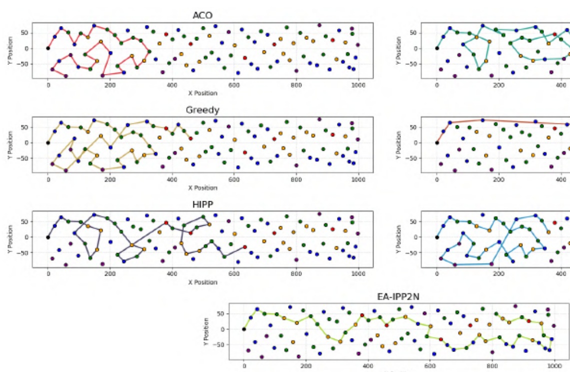
Figure 4.3 Heatmap of Scenario 2 (SC 2), illustrating priority levels where warmer colors represent areas of higher importance for mission planning.



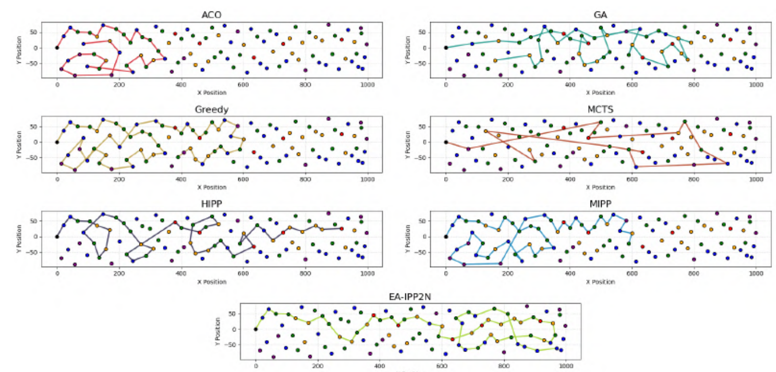
(a) Scenario 1 Mission 2



(b) Scenario 1 Mission 5



(c) Scenario 2 Mission 1



(d) Scenario 2 Mission 3

Figure 4.4 Planned paths in different scenarios. Higher node reward values are indicated by warmer colors.

Across the five missions in SC1, EA-IPP2n outperformed HIPP by 14–41%, MIPP by 10–37%, GA by 2–13%, ACO by 34–52%, Greedy by 1–24%, and MCTS by 57–67% in total rewards. EA-IPP2n consistently adhered to both the energy and time budgets, demonstrating its effectiveness for constrained UAV missions.

In SC2, EA-IPP2n led in total rewards in all missions, surpassing HIPP by 0.7–23%, MIPP by 23–71%, GA by 3–25%, ACO by 21–49%, Greedy by 16–41%, and MCTS by 57–77%. While Greedy occasionally achieved rewards close to EA-IPP2n, it exceeded the energy budget by up to 14% and incurred significantly higher turning effort (up to 70%), limiting its suitability for budget-constrained missions.

The proposed EA-IPP2n algorithm achieved the highest total rewards across all evaluated scenarios, with a consistent average performance of (24.9 ± 3.6), indicating both high effectiveness and low variability across the 10 missions. It outperformed all benchmark algorithms: GA (21.2 ± 4.0), Greedy (19.7 ± 4.2), MIPP (18.8 ± 4.1), HIPP (20.3 ± 5.1), ACO (13.8 ± 5.0), and MCTS (6.9 ± 1.1) across both evaluated scenarios (10 missions), as illustrated in Fig. 4.5.

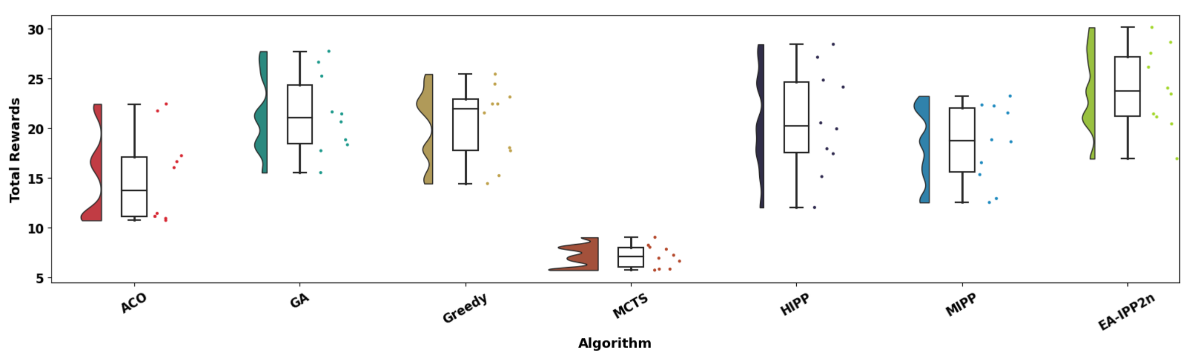


Figure 4.5 Raincloud plot of Total Rewards collected by each algo across missions (SC1 & SC2).

Pairwise comparisons confirm meaningful gains: +9.3 over ACO (+66%), +5.7 over MIPP (+33%), +3.0 over HIPP (+18%), +3.4 over Greedy (+18%), and +2.7 over GA (+11%). The most substantial improvement was observed over MCTS, with an average gain of +17.8, representing more than three times the performance. Notably, EA-IPP2n achieved these results without exceeding energy or time constraints, unlike GA, Greedy, and MCTS, which violated energy limits in several missions. These findings demonstrate that EA-IPP2n offers superior reward efficiency, consistent performance, and robust compliance with dual operational constraints, confirming its effectiveness for energy- and time-constrained UAV missions.

Furthermore, EA-IPP2n maintained high node coverage while respecting budgets. Relative to MCTS, EA-IPP2n visited $\sim$3–4 times more nodes across all missions (e.g., Scenario 1, Mission 1: 38 vs. 13). It also surpassed ACO by 20–60% in most missions (e.g., Scenario 2, Mission 3: 52 vs. 31), remaining comparable only at the largest budget (Scenario 1, Mission 5: 64 vs. 57). EA-IPP2n visited more nodes in the majority of cases against HIPP, MIPP, and GA; the main exception was Scenario 1, Mission 4 (Table 4.6), where MIPP reached 54 nodes versus 52 nodes for EA-IPP2n, yet EA-IPP2n still achieved a higher total reward (24.0 vs. 21.5 for MIPP). Greedy occasionally attained the highest node count, but this coincided with energy overruns of $\approx$6–14% and significantly higher turning effort, as seen in the Energy and Turning Angles columns (e.g., Scenario 1, Mission 3: 327.7 kJ vs. 288 kJ; 3709° vs. 2186°). EA-IPP2n achieved effective area coverage while fully adhering to the operational budgets, avoiding the energy excesses and frequent turns exhibited by Greedy.

On the other hand, EA-IPP2n achieved near instantaneous processing times of under 0.015 s across all missions, being more than 2 times faster than HIPP, up to 30 times faster than MIPP, and several hundred times faster than GA, ACO, and MCTS. While Greedy Search remained the fastest (0.002 s), it consistently violated the energy budget, whereas EA-IPP2n respected all constraints while still maintaining competitive computational efficiency.

4.5.4 MISSION PERFORMANCE SCORE

To provide a comprehensive and fair evaluation of all tested algorithms, we designed a mission performance score (MPS) that consistently integrates heterogeneous metrics. The score accounts for both benefits (collected rewards) and costs (energy, completion time, and processing time), while explicitly enforcing mission budgets.

For a given algorithm A in mission m, the score is defined as:

$$\mathrm{MPS}(A,m) = \sigma R_{A,m}^{\text{norm}} + \xi E_{A,m}^{\text{inv.norm}} + \delta T_{A,m}^{\text{inv.norm}} + \tau PT_{A,m}^{\text{inv.norm}} \tag{4.10}$$

where:

- $R_{A,m}^{\text{norm}}$ is the normalized reward, computed as the ratio of the reward collected by algorithm A in mission m to the maximum achievable reward in the corresponding scenario.
- $T_{A,m}^{\text{inv.norm}}$ is the inverted normalized completion time, defined as $1 - \frac{T_{A,m}}{T_m^{\max}}$, where $T_m^{\max}$ is the mission-specific time budget. Values below the budget contribute positively, while exceeding the budget yields negative contributions.
- $E_{A,m}^{\text{inv.norm}}$ is the inverted normalized energy consumption, defined as $1 - \frac{E_{A,m}}{E_m^{\max}}$, where $E_m^{\max}$ is the mission-specific energy budget. As with time, exceeding the budget produces a penalty (negative value).
- $PT_A^{\text{inv.norm}}$ is the inverted normalized average processing time of algorithm A in mission m. It is obtained by linearly mapping average processing times to $[0,1]$ and then inverting them so that faster algorithms receive higher contributions.

Normalization ensures that heterogeneous quantities are placed on a comparable scale. For cost-related terms (energy, time, and processing time), inversion ensures that lower values translate into higher contributions, i.e., inv.norm $= 1 -$ norm. Crucially, when a cost exceeds its corresponding budget, the normalized value becomes greater than 1, producing a negative inverted score. This mechanism directly penalizes over-budget or infeasible solutions, preventing them from appearing competitive due to high rewards alone.

The weighting coefficients are set to $\sigma = 0.4$ (reward), $\xi = 0.3$ (energy consumption), $\delta = 0.2$ (completion time), and $\tau = 0.1$ (processing time). These values reflect the relative importance of each factor and ensure that excessive costs cannot be fully compensated by rewards.

For each mission m, the score is computed independently in SC1 and SC2, and the final mission-level value is obtained by averaging the two. This procedure yields a robust measure of performance across different environments. Detailed per-mission results are presented in Tables 4.8 and 4.9, and the performance score distributions are visualized in Fig. 4.6 across 10 missions (SC1 + SC2).

Table 4.8 Mission-level performance scores for all algorithms in Scenario 1 (SC1). **Bold**: best, <u>underline</u>: second best.

Algorithm	Mission 1	Mission 2	Mission 3	Mission 4	Mission 5
ACO	0.2056	0.2544	0.2520	0.2941	0.3197
GA	0.2297	0.2852	0.2662	0.2937	0.3492
Greedy	0.2298	0.2657	0.2979	0.3399	0.3319
MCTS	0.1341	0.1564	0.1675	0.1530	0.1765
HIPP	0.2570	0.2881	0.3199	0.3496	<u>0.3939</u>
MIPP	<u>0.2669</u>	<u>0.2957</u>	<u>0.3307</u>	<u>0.3604</u>	0.3702
EA-IPP2n	**0.3028**	**0.3415**	**0.3363**	**0.3769**	**0.4444**

Table 4.9 Mission-level performance scores for all algorithms in Scenario 2 (SC2). **Bold**: best, underline: second best.

Algorithm	Mission 1	Mission 2	Mission 3	Mission 4	Mission 5
ACO	0.2283	0.2446	0.2776	0.2833	0.3144
GA	0.2429	0.2518	0.2811	0.3133	0.3424
Greedy	0.2300	0.2591	0.2951	0.3347	0.3286
MCTS	0.1320	0.2035	0.1618	0.1505	0.1546
HIPP	0.3006	0.3273	0.3725	**0.4017**	**0.4162**
MIPP	0.2560	0.2993	0.3226	0.3557	0.3670
EA-IPP2n	**0.3434**	**0.3646**	**0.3838**	0.3990	0.4126

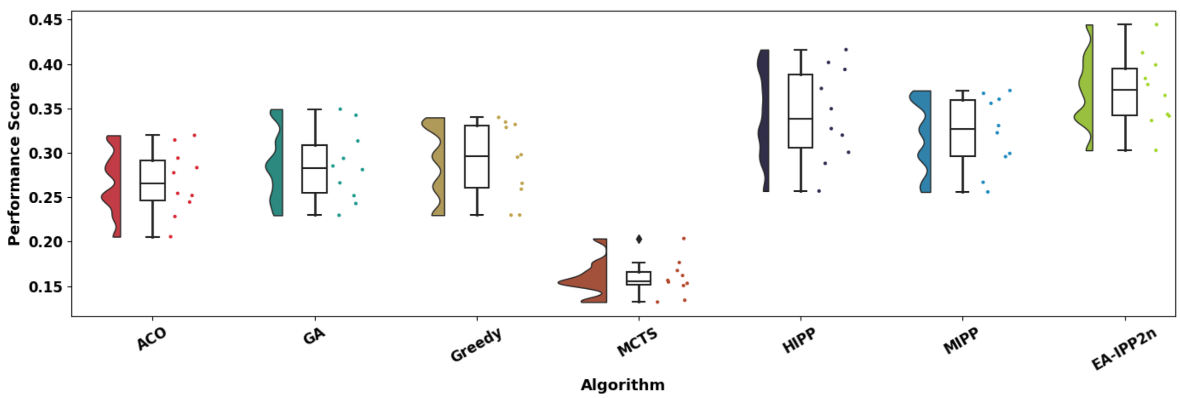


Figure 4.6 Raincloud plot of algorithm performance across 10 missions in SC1 & SC2.

As shown in Tables 4.8 and 4.9, EA-IPP2n consistently achieves the highest performance score across 8 out of 10 missions. HIPP and MIPP follow as the closest competitors, with HIPP slightly outperforming MIPP. GA and Greedy occupy the mid-range, while ACO performs slightly lower overall. MCTS consistently yields the lowest scores, reflecting limited suitability for the considered tasks.

On average across all missions and scenarios, EA-IPP2n attains a mean score of 0.380, compared with 0.343 for HIPP and 0.323 for MIPP. This represents an improvement of approximately 11% over HIPP and 17.6% over MIPP. Relative to the classical baselines, EA-IPP2n outperforms GA and Greedy by 35.2% and 31.9%, respectively, while surpassing

ACO by 41.8%. The weakest competitor, MCTS, averages 0.164, which is 57% lower than EA-IPP2n.

The raincloud visualization in Fig. 4.6 provides further insight into the distribution of mission-level scores. EA-IPP2n not only attains the highest median score but also exhibits relatively low variability, confirming both robust and consistent performance. HIPP and MIPP show competitive yet more variable outcomes, while GA, Greedy, and ACO form a mid-performing group. MCTS displays consistently poor and tightly clustered low scores.

4.6 SUMMARY

In this Chapter 4, we presented EA-IPP2n, an Energy-Aware IPP algorithm tailored for UAV operations in time-critical and resource-limited environments such as SAR missions. The proposed method introduces a multi-factor cost framework, modeling translational and rotational energy consumption alongside reward value for more realistic UAV planning. EA-IPP2n functions as a pre-mission planner, computing the full UAV path prior to deployment to ensure optimal resource utilization and mission efficiency. By incorporating a two-hop look-ahead strategy and a reward-normalized cost function, EA-IPP2n effectively balances energy, time, and information gain during path selection.

Simulations show that EA-IPP2n outperforms established metaheuristic algorithms in reward collection, turning effort, and constraint satisfaction. Unlike many baseline methods, EA-IPP2n adheres to both energy and time budgets across all scenarios. The performance score highlights the effectiveness of EA-IPP2n, which achieved the highest overall value (0.380 ± 0.041 on average, ranging from 0.323 to 0.429 across missions), with relative improvements between 11% and 57% compared with the other algorithms.

Furthermore, EA-IPP2n demonstrated complete adherence to both time and energy budgets across all missions. Although the time budget constraint was uniformly applied to all algorithms, only the dual-budget formulation of EA-IPP2n, HIPP, and MIPP ensured full compliance with both constraints. In contrast, Greedy and GA consistently violated the energy budget, MCTS exceeded it in 40% of missions, and ACO, despite its heuristic nature, maintained full energy compliance. Overall, EA-IPP2n was the sole algorithm that achieved complete dual-budget compliance while simultaneously delivering the highest total reward performance.

CHAPTER V

VAP & EA-MIPP: V-SHAPE AREA PARTITIONING AND ENERGY-AWARE MULTI-UAV INFORMATIVE PATH PLANNING WITH NFZ AVOIDANCE

5.1 OVERVIEW

Multi-UAV missions have become essential for addressing large-scale and complex tasks such as wide-area search and rescue, disaster response, and environmental monitoring. However, multi-UAV missions come with significant challenges, including risks of collision, overlapping or redundant trajectories, communication and coordination requirements, and the need for balanced workload distribution among UAVs.

Within this broader context, Multi-UAV Informative Path Planning (MIPP) has emerged as a key approach for exploration, mapping, surveillance, and multi-target search. However, MIPP inherits the coordination complexities of multi-UAV systems, which makes ensuring efficiency and safety challenging without the support of additional planning strategies. A widely studied solution is *area partitioning*, where the mission space is divided among UAVs to balance workloads and minimize redundancy. Yet, in the case of MIPP, the challenge is more complex: instead of simply dividing the overall Area of Interest (AOI), the partitioning must focus on Regions of Interest (ROIs), which may be irregularly distributed or overlapping. AOI refers to the overall geographical region to be covered by the UAV team, while ROI's are specific sub-areas within the AOI that contain high-value information or tasks. Moreover, practical deployments must also account for NFZ and obstacle avoidance and, adding further complexity to cooperative mission planning.

To address these challenges, we propose VAP & EA-MIPP, a framework for *V-shape Area Partitioning and Energy-Aware Multi-UAV Informative Path Planning with NFZ Avoidance*. The key contributions are summarized as follows:

- **VAP:** A V-shape partitioning method that allocates Regions of Interest (ROIs) among UAVs, ensuring balanced workload distribution and minimizing redundant coverage (overlapping).
- **EA-MIPP:** An extension of our previous EA-IPP2n (Chapter 4) to a cooperative multi-UAV setting, incorporating NFZ and 2D static obstacle avoidance (buildings, trees, etc.) into informative path planning to ensure safe and feasible missions.

The proposed framework provides a scalable, energy-aware solution for cooperative UAV missions. Multiple UAVs collectively maximize information gain while respecting mission constraints and maintaining safe operation under real-world conditions

5.2 BACKGROUND

Cooperative coverage and area partitioning for multi-UAV systems have been widely investigated, motivated by applications in surveillance, reconnaissance, monitoring, and mapping. Existing research can broadly be categorized into three groups: *(i)* geometric and clustering-based partitioning for workload balancing, *(ii)* optimization-driven coverage path planning, and *(iii)* informative area exploration and multi-UAV coordination.

5.2.1 GEOMETRIC AND CLUSTERING-BASED PARTITIONING

Early work emphasized geometric decomposition and distributed coordination to balance sub-areas. Ann *et al.* [274] modeled multi-UAV coverage as a Multiple Traveling Salesman

Problem (MTSP), introducing clustering and graph-cut based allocation to achieve collision-free partitions. Balampanis *et al.* [275] employed Constrained Delaunay Triangulation for partitioning complex coastal regions, accounting for UAV capabilities and field-of-view. Acevedo *et al.* [276] proposed a distributed approach based on neighbor-to-neighbor communication, achieving balanced partitions under communication constraints. Xiao *et al.* [10] advanced this line of work with power diagrams (weighted Voronoi) to maintain compact and adaptive subregions in persistent monitoring tasks. While effective for scalability and robustness, these approaches typically focus on uniform partitioning of the AOI rather than ROI's, limiting their applicability in informative missions.

5.2.2 OPTIMIZATION-BASED COVERAGE PLANNING

Optimization methods enhance partitioning quality and coverage efficiency by considering workload balance, UAV heterogeneity, and energy limitations. Priyadarsini *et al.* [277] introduced rectilinear partitioning optimized with Particle Swarm Optimization and Firefly Algorithms. Chen *et al.* [278] proposed Weighted Balanced Graph Partitioning (WBGPP), combining workload balancing with intra-region single-agent coverage path optimization. Zhang *et al.* [279] addressed energy-constrained monitoring via a "plan-then-divide" strategy that integrates team-level orienteering with subregion coverage allocation. Gao *et al.* [280] introduced a hierarchical explore–exploit framework for large infrastructure inspection, using density-guided Voronoi partitioning for exploration followed by TSP-based local inspection tours. These contributions improve efficiency and workload balancing, particularly in heterogeneous or resource-constrained scenarios, yet remain primarily oriented toward complete coverage rather than ROI-driven informative missions.

5.2.3 INFORMATIVE AREA EXPLORATION AND ADAPTIVE PARTITIONING

Informative area exploration has become an active research direction, with the objective of maximizing information gain or reducing uncertainty. Gui *et al.* [281] proposed a decentralized centroid-based partitioning method coupled with receding-horizon next-best-view planning for rapid 3D exploration. Cho *et al.* [48] combined Gaussian-process-based sensing with wind-aware routing to optimize informative field mapping. Jati *et al.* [282] extended informative coverage to heterogeneous UAV–UGV teams for contamination mapping, prioritizing regions via weighted Voronoi tessellations. Tian *et al.* [283] tackled dynamic livestock monitoring using streaming K-means clustering and reinforcement learning for real-time data collection.

Orisatoki [258] introduced the Information-theoretic Heuristic Informative Path Planning (HIPP) algorithm for mapping unknown non-convex environments using occupancy grid maps and LiDAR-based exploration. HIPP estimates mapping gain heuristically for frontier cells and employs an ε-greedy strategy to balance exploration and exploitation. To ensure safe navigation in cluttered maps, HIPP integrates a sampling-based obstacle avoidance mechanism: whenever a direct path to the next information-rich region is blocked, an RRT* module generates a collision-free trajectory. However, while robust in exploration and obstacle avoidance, HIPP does not explicitly consider UAV-specific constraints such as energy consumption, turning costs, or flight dynamics, which limits its applicability in time and energy critical aerial missions.

These approaches emphasize adaptivity and uncertainty reduction, but many are limited in addressing strict mission constraints such as energy budgets, NFZs, and heterogeneous coordination.

Prior research demonstrates that geometric partitioning ensures scalability, optimization-based methods improve efficiency and energy balance, and informative frameworks adapt to uncertainty and dynamic missions. However, most existing solutions focus on AOI-level partitioning and uniform coverage, often overlooking the explicit handling of ROIs in informative tasks. Furthermore, NFZ and obstacle avoidance, critical for real-world deployments, are rarely integrated with informative path planning.

5.3 V-SHAPE AREA PARTITIONING AND MULTI-UAV INFORMATIVE PATH PLANNING

5.3.1 PROBLEM OVERVIEW

In multi-UAV missions such as SAR, the AOI represents the global mission space (e.g., a disaster zone). However, not all parts of the AOI are relevant. Instead, responders prioritize ROIs, which denote areas with higher likelihood of containing survivors, equipment, or mission-critical information. ROIs may be inferred from heterogeneous sources such as thermal signatures, mobile phone signals, or object detectors. Focusing on ROIs ensures that UAV resources are allocated to the most valuable areas rather than uniformly covering the entire AOI.

At the same time, deploying multiple UAVs introduces coordination challenges: (i) overlapping or redundant trajectories, (ii) unbalanced workloads, and (iii) safety risks due to NFZs or static obstacles (e.g., buildings, trees). To address these challenges, we propose a two-stage approach: 1) *V-shape Area Partitioning (VAP)* divides the ROI into balanced angular partitions emanating from the starting point, ensuring balanced workload allocation and spatial separation of UAV trajectories. 2) *Energy-Aware Multi-UAV Informative Path Planning with*

Obstacle and NFZ Avoidance (EA-MIPP) generates efficient, reward-maximizing trajectories within each partition while respecting energy, time, and safety constraints.

Figure 5.1 illustrates an example of AOI heatmap used as initial mission input.

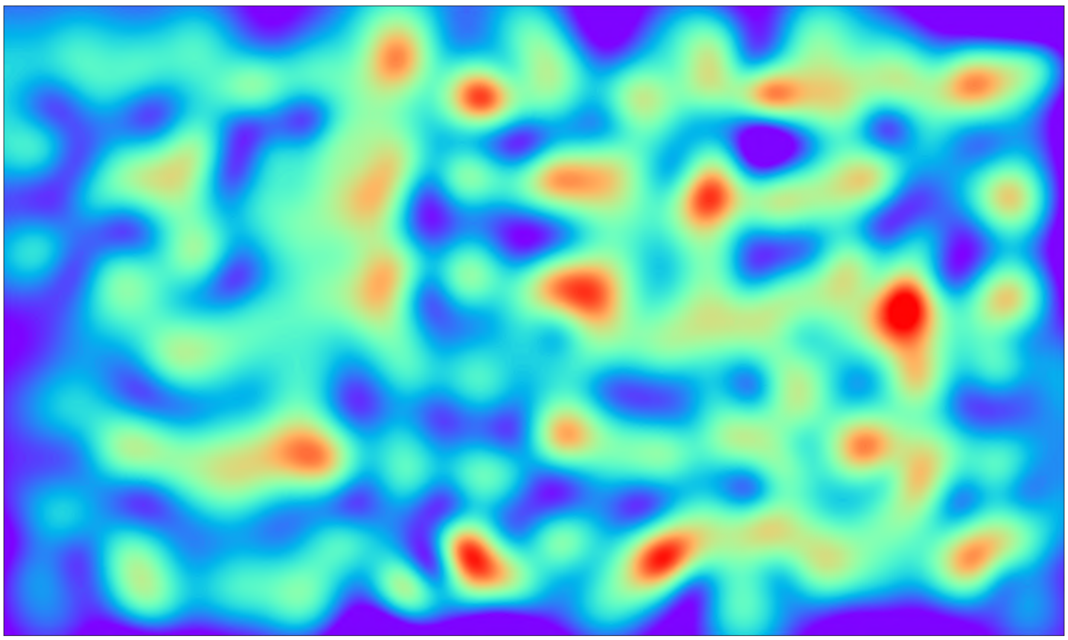


Figure 5.1 Initial AOI heatmap visualized as spatial importance data for a multi-UAV mission.

5.3.2 ROI DETECTION FROM AOI

The AOI heatmap $I(x,y)$ encodes spatial importance, derived from semantic, thermal, or communication signals. In this work, ROIs are extracted using the Ultralytics YOLO11 detector [284]. For controlled experiments, a color-based HSV mask serves as a deterministic baseline.

Each detection by YOLO is represented as a tuple

$$d_i = (b_i, c_i, s_i), \tag{5.1}$$

where b_i is the bounding box, c_i is the predicted class, and s_i is the confidence score. Not all pixels within b_i are ROI pixels. To identify only the relevant pixels within each detection b_i, we define a binary mask $M_i(x,y) \in \{0,1\}$ where pixels belonging to the ROI are marked with 1 and others with 0. The final ROI mask is obtained by performing a pixel-wise logical OR across all detections:

$$M(x,y) = \bigvee_{i=1}^{N_{dt}} M_i(x,y), \tag{5.2}$$

where N_{dt} is the total number of detections, and each mask $M_i(x,y)$ is defined as:

$$M_i(x,y) = \begin{cases} 1, & \text{if pixel } (x,y) \text{ belongs to the ROI of detection } i, \\ 0, & \text{otherwise.} \end{cases}$$

The set of all ROI pixels is defined as:

$$\Omega_{\text{ROI}} = \{(x,y) \in \Omega \mid M(x,y) = 1\}, \tag{5.3}$$

where Ω denotes the global area of interest (AOI), and $\Omega_{\text{ROI}} \subseteq \Omega$ contains the pixels identified as part of any ROI.

The total area covered by all detected ROIs is given by $A(ROI)_{\text{tot}}$ which represents the cumulative area of all detected regions of interest.:

$$A(ROI)_{\text{tot}} = \iint_{\Omega} M(x,y)\, dx\, dy = \iint_{\Omega_{\text{ROI}}} 1\, dx\, dy, \tag{5.4}$$

Figure 5.2 shows detected ROIs in a sample AOI heatmap, highlighting the refinement of the initial mission space into meaningful regions.

In a cooperative mission, it is essential to distribute the detected ROIs fairly among UAVs to prevent redundant coverage and ensure a balanced workload. To achieve this, we introduce a partitioning strategy that divides AOI in a way that ensures each UAV is assigned a region containing an approximately equal share of the total ROI area. This allows for balanced workload distribution across UAVs.

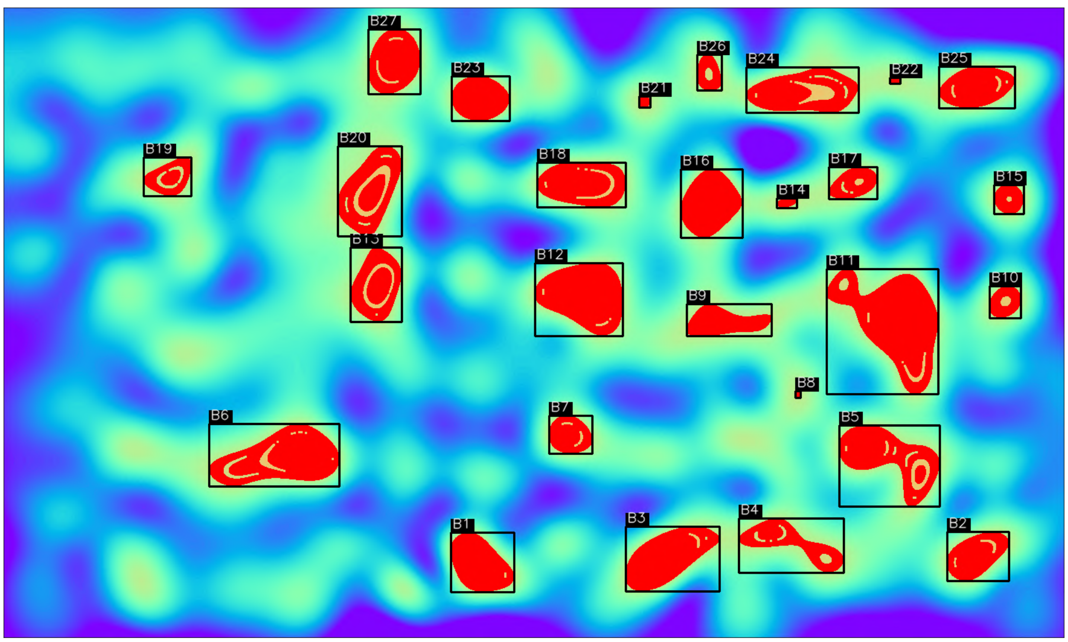


Figure 5.2 Detected ROIs (red) within the AOI heatmap using YOLO11 detections combined with binary mask segmentation.

5.3.3 V-SHAPE EQUAL-AREA PARTITIONING

This step divides the AOI into balanced angular subregions, one for each UAV. Unlike simple geometric division, the partitioning is guided by the ROI distribution within the AOI, ensuring that UAV workloads are balanced according to the actual importance of detected regions. In other words, the ROI distribution determines how the AOI is partitioned.

PARTITION CONCEPT

Let $U = \{u_1, \ldots, u_m\}$ denote the UAV team, and let $n_0 = (x_0, y_0)$ be their common starting point. The environment is modeled as a graph whose nodes represent locations of interest, each associated with a reward value (as described in Chapter 4). Each point (x, y) in the AOI is represented in polar coordinates relative to n_0 as:

$$\theta(x,y) = \operatorname{atan2}(y - y_0, x - x_0), \quad \theta \in (-\pi, \pi]. \tag{5.5}$$

Here, $\theta(x,y)$ is the angular direction of (x,y) with respect to the UAV start point.

Each detected ROI contributes a workload proportional to its area. Formally, we denote this by $w_{\mathrm{ROI}}(x,y)$, which in the current implementation equals the ROI's pixel area. The *weighted area* of the j-th partition is then given by:

$$A_j(\Phi) = \iint_{\Omega_j(\Phi)} w_{\mathrm{ROI}}(x,y)\, dx\, dy, \tag{5.6}$$

where $\Phi = \{\phi_1, \ldots, \phi_{m-1}\}$ denotes the set of angular cut boundaries that divide the AOI into m V-shaped sectors. Here, $w_{\mathrm{ROI}}(x,y) \in [0,1]$ generalizes the binary ROI mask $M(x,y)$ by assigning continuous importance values, with

$$w_{\mathrm{ROI}}(x,y) = I(x,y) \cdot M(x,y),$$

where $I(x,y)$ is the AOI heatmap intensity and $M(x,y)$ is the ROI membership mask. Then the total weighted area is:

$$A(ROI)_{\text{tot}} = \sum_{j=1}^{m} A_j(\Phi)$$

If all weights are equal, i.e., $w_{\text{ROI}}(x,y) = 1$, the formulation reduces to a purely geometric division. When ROI areas are considered, the effective workload of each partition is measured as the cumulative sum of ROI contributions. Let the detected ROIs be indexed in ascending order of angle as $(x_1,y_1),(x_2,y_2),\ldots,(x_N,y_N)$. The cumulative workload after traversing the first l ROIs is defined as

$$C(l) = \sum_{i=1}^{l} w_{\text{ROI}}(x_i,y_i), \tag{5.7}$$

where ROIs are processed in order of increasing polar angle θ.

Thus, UAVs are assigned angular partitions such that each one is responsible for approximately the same fraction of the total workload:

$$A_{\text{target}} = \frac{A(ROI)_{\text{tot}}}{m}. \tag{5.8}$$

ROI-GUIDED PARTITIONING OF THE AOI

The algorithm uses the ROI distribution to determine where to place the angular cut boundaries. The procedure is as follows:

- Compute the polar angle $\theta(x,y)$ for all ROI centroids relative to the starting point n_0.
- Sort all ROI points in ascending order of θ (clockwise around n_0).
- Accumulate the ROI areas $w_{\text{ROI}}(x,y)$ as the ROIs are traversed.

- Whenever the cumulative workload first exceeds the next threshold value A_{target}, $2A_{\text{target}}$, $3A_{\text{target}}, \ldots$, insert a new boundary angle ϕ_j between the previous and current ROI points.
- If the ROI distribution is too sparse or uneven to define all cuts, add evenly spaced fallback angles to ensure complete 360° coverage.

This approach ensures that the AOI is divided directly according to the ROI distribution. Each angular sector is defined as:

$$\Omega_j(\Phi) = \{(x,y) \in \Omega_{\text{AOI}} \mid \phi_{j-1} < \theta(x,y) \leq \phi_j\}, \quad j = 1, \ldots, m, \tag{5.9}$$

with boundary conditions $\phi_0 = -\pi$ and $\phi_m = \pi$.

BALANCED WORKLOAD OBJECTIVE

To ensure equitable workload among UAVs, the algorithm minimizes the variance of weighted ROI areas across all partitions:

$$\min_{\Phi} J_{\text{area}}(\Phi) = \sum_{j=1}^{m} \left(A_j(\Phi) - \frac{A(ROI)_{\text{tot}}}{m} \right)^2, \tag{5.10}$$

Minimizing $J_{\text{area}}(\Phi)$ ensures that each UAV receives a partition containing approximately the same workload, achieving balanced collaboration and efficient coverage.

Algorithm 5.1 summarizes the procedure, while Figure 5.3 shows an example partitioning for $m = 3$ UAVs.

Input: ROI points (x,y) with weights $w_{\mathrm{ROI}}(x,y)$, UAV start point n_0, number of UAVs m
Output: Cut angles $\Phi = \{\phi_1, \ldots, \phi_{m-1}\}$ and partitions $\Omega_j(\Phi)$

Compute polar angle $\theta(x,y)$ for all ROI points using Eq. (5.5).
Sort ROI points in ascending order of $\theta(x,y)$.
Compute total weighted area $A(ROI)_{\mathrm{tot}}$ and per-UAV target $A_{\mathrm{target}} = A(ROI)_{\mathrm{tot}}/m$.
Initialize cumulative weighted area $C = 0$, threshold $T = A_{\mathrm{target}}$, and set $\Phi = \emptyset$.
foreach *ROI point* (θ, w_{ROI}) *in sorted order* **do**
    Update $C \leftarrow C + w_{\mathrm{ROI}}$.
    if $C \geq T$ *and* $|\Phi| < m-1$ **then**
        Insert cut angle ϕ midway between the previous and current θ.
        Update $T \leftarrow T + A_{\mathrm{target}}$.
    end
end
if $|\Phi| < m-1$ **then**
    Add evenly spaced fallback cuts until $|\Phi| = m-1$.
end
Set $\phi_0 = -\pi$ and $\phi_m = \pi$.
Assign each AOI point to a partition as defined in Eq. (5.9).

Algorithm 5.1: V-shape Equal-Area Partitioning

5.3.4 EA-MIPP: ENERGY-AWARE MULTI-UAV INFORMATIVE PATH PLANNING

After partitioning, each UAV u_j operates within its assigned partition Ω_j. Path planning in each partition is formulated as a localized Orienteering Problem, where the objective is to maximize the accumulated reward R under both time and energy constraints (Chapter 4.3):

$$\max_{p_j} R(p_j) \quad \text{s.t.} \quad T(p_j) \leq T_{\mathrm{bud}},\ E(p_j) \leq E_{\mathrm{bud}}.$$

Nodes located inside obstacles or NFZs are removed, and edges intersecting such regions are pruned. Each UAV then executes an energy-aware two-step lookahead search adapted from EA-IPP2n (Sec. 4.4). At each expansion, candidate transitions $(i \to j \to k)$ are evaluated by combining normalized time (N_T), energy (N_E), and average reward $(\bar{r}_{kj})$ through the cost

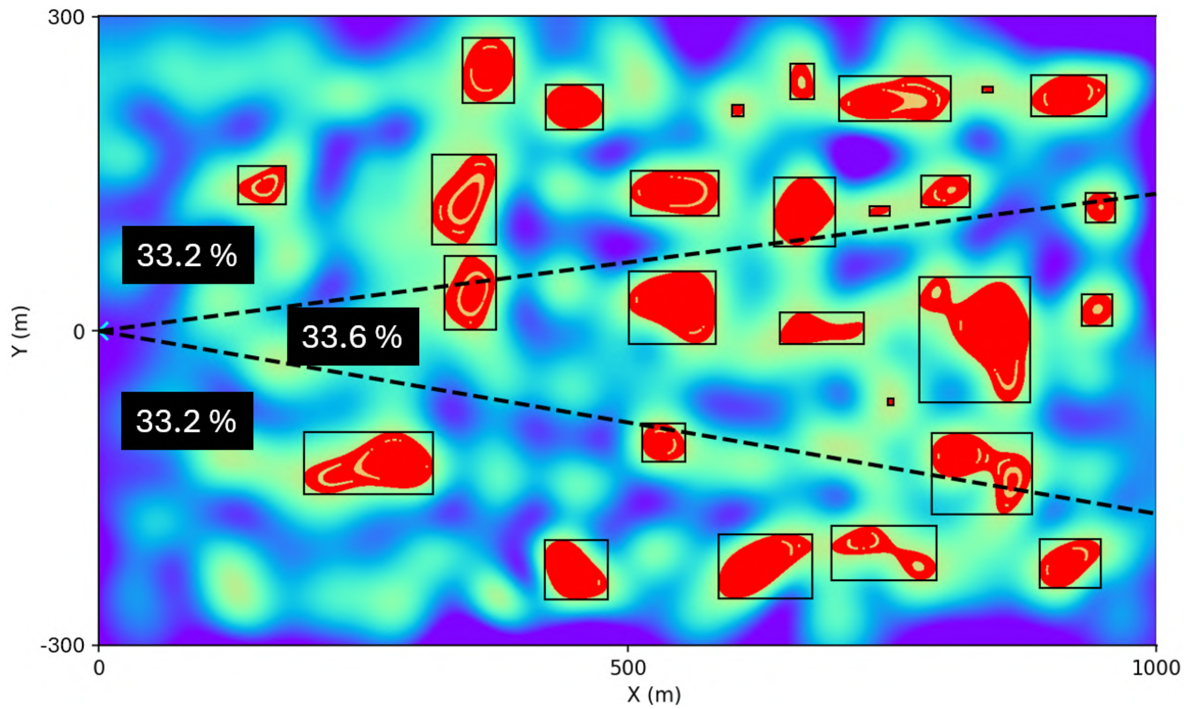


Figure 5.3 Example of V-shape partitioning into 3 UAV sectors with balanced ROI-based workload allocation.

function

$$C_{kj} = \alpha N_T + \alpha N_E + \beta(1 - \bar{r}_{kj}) \text{ (Eq. 4.8).}$$

The planner selects the transition that maximizes the expected reward gain while maintaining feasibility within the time and energy budgets $(T_{\text{bud}}, E_{\text{bud}})$. The process iteratively expands the most promising sequence of nodes until no feasible extension remains, thus producing an energy- and reward-optimal path within each partition.

This approach effectively extends EA-IPP2n to multi-UAV coordination by running the planning process independently within each partition Ω_j obtained from the V-shape Equal-Area Partitioning (VAP). Figure 5.4 illustrates representative paths obtained for three UAVs using EA-MIPP within the partitioned AOI.

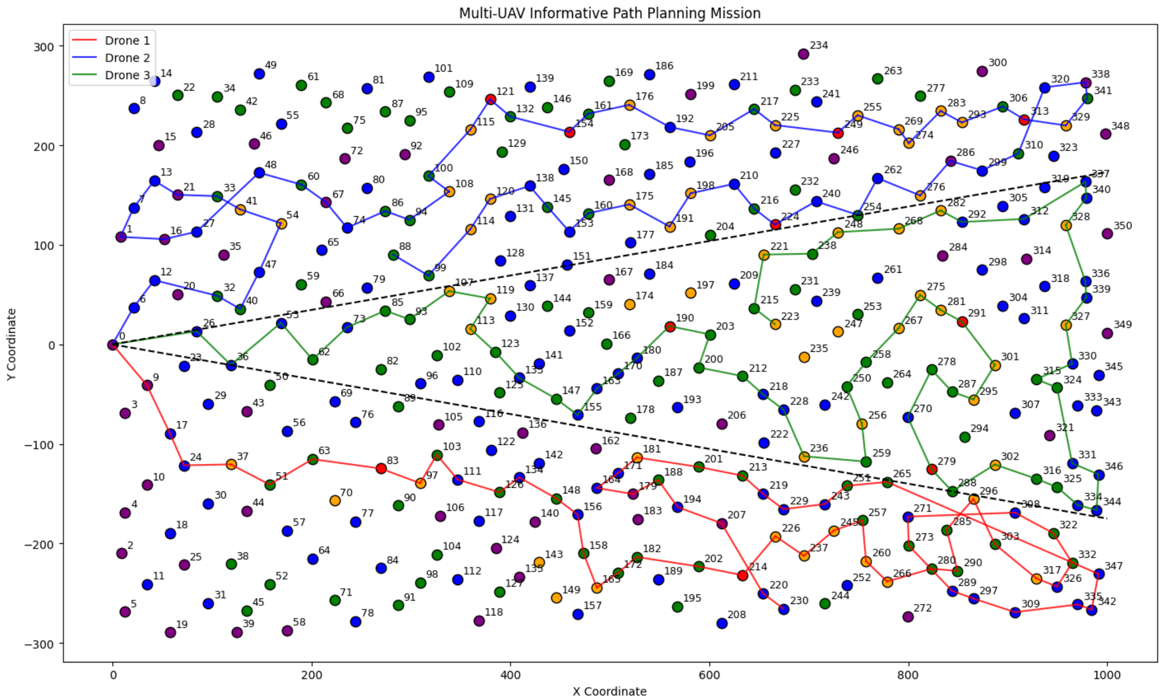


Figure 5.4 EA-MIPP applied within each partitioned area for a 3-UAV mission.

5.4 EVALUATION AND RESULTS

The effectiveness of the proposed VAP-MIPP framework was assessed through five multi-UAV missions, each executed with three UAVs operating under identical motion parameters. Mission budgets were progressively increased, ranging from $T_{\text{bud}} = 900$ s and $E_{\text{bud}} = 216$ kJ in Mission 1 to $T_{\text{bud}} = 1500$ s and $E_{\text{bud}} = 360$ kJ in Mission 5. The budgets were scaled proportionally to Mission 5 using factors of 0.6, 0.7, 0.8, 0.9, and 1.0, respectively, corresponding to endurance windows of 15–25 minutes. These settings reflect the capabilities of rotary-wing UAVs typically employed in SAR and mapping missions. For instance, a six-cell Li-Po battery with 4500 mAh capacity at 22.2 V stores approximately 359 kJ:

$$E_{\text{total}} = 22.2\,\text{V} \times 4.5\,\text{Ah} \times 3600\,\text{s/h} \approx 359\,\text{kJ}. \tag{5.11}$$

Hence, Mission 5 represents a realistic upper bound, while Missions 1–4 emulate increasingly constrained operational conditions. All simulations were conducted on *Ubuntu 20.04*, running on an *Intel i7-6700HQ CPU*, *16 GB RAM*, and an *NVIDIA GTX 965M GPU*.

All missions were simulated within a fixed 1000×600 m environment containing 350 distributed nodes, as shown in Fig. 5.1. UAV forward speed was set to $v = 2.0\,\text{m/s}$ to balance stability and maneuverability, while rotation rate was fixed at $\omega = 60°/\text{s}$. The energy consumption per unit distance, $\lambda = 0.1164\,\text{kJ/m}$, and per degree $\gamma = 0.0173\,\text{kJ/degree}$ [262].

For fair comparison, the AOI was partitioned using VAP in all experiments, and two path planning approaches were tested: EA-MIPP (our method) and HIPP (baseline [258]). Both planners were evaluated under the same dual-constraint model accounting for time and energy consumption. To ensure fairness, HIPP was extended to explicitly respect the energy budget. Results for Missions 1–5 are summarized in Tables 5.1–5.5.

Table 5.1 Performance comparison of VAP & MIPP and HIPP for Mission 1 (multi-UAV case with 3 UAVs).$T_{\text{bud}} = 900\,\text{s}$ and $E_{\text{bud}} = 216\,\text{kJ}$.

Algorithm	Metrics	UAV 1	UAV 2	UAV 3	Total
EA-MIPP	Path Length (m)	1537.39	1517.15	1565.98	4620.52
	Turning Angles (°)	1917.59	2067.46	1845.16	5830.21
	Completion Time (s)	800.65	793.03	813.74	2407.42
	Energy Consumption (kJ)	212.13	212.36	214.20	638.69
	Visited Nodes	39	41	41	**121**
	Total Rewards	12.20	14.12	17.70	**44.02**
	Process Time (s)				**0.9197**
HIPP	Path Length (m)	1675.32	1675.18	1682.20	5032.70
	Turning Angles (°)	1036.91	1074.57	931.97	3043.45
	Completion Time (s)	854.94	855.50	856.63	2567.07
	Energy Consumption (kJ)	212.95	213.58	211.93	638.46
	Visited Nodes	38	37	37	112
	Total Rewards	13.70	12.20	13.60	39.50
	Process Time (s)				1.1777

Table 5.2 Performance comparison of VAP & EA-MIPP and HIPP for Mission 2 (multi-UAV case with 3 UAVs). $T_{\text{bud}} = 1050\,\text{s}$ and $E_{\text{bud}} = 252\,\text{kJ}$.

Algorithm	**Metrics**	**UAV 1**	**UAV 2**	**UAV 3**	**Total**
EA-MIPP	Path Length (m)	1714.86	1748.33	1790.48	5253.67
	Turning Angles (°)	2139.13	2237.94	2501.51	6878.58
	Completion Time (s)	893.08	911.46	936.93	2741.47
	Energy Consumption (kJ)	236.62	242.22	251.69	730.53
	Visited Nodes	43	47	47	**137**
	Total Rewards	13.40	17.42	21.40	**52.22**
	Process Time (s)				**1.0411**
HIPP	Path Length (m)	1968.53	1975.12	1972.57	5916.22
	Turning Angles (°)	1164.98	1203.10	1156.95	3525.03
	Completion Time (s)	1003.68	1007.61	1005.57	3016.86
	Energy Consumption (kJ)	249.29	250.72	249.62	749.63
	Visited Nodes	45	43	43	131
	Total Rewards	15.20	14.70	16.00	45.90
	Process Time (s)				1.3707

Table 5.3 Performance comparison of VAP & EA-MIPP and HIPP for Mission 3 (multi-UAV case with 3 UAVs). $T_{\text{bud}} = 1200\,\text{s}$ and $E_{\text{bud}} = 288\,\text{kJ}$.

Algorithm	**Metrics**	**UAV 1**	**UAV 2**	**UAV 3**	**Total**
EA-MIPP	Path Length (m)	2071.23	1999.67	2023.11	6094.01
	Turning Angles (°)	2395.56	2737.53	2818.89	7951.98
	Completion Time (s)	1075.54	1045.46	1058.54	3179.54
	Energy Consumption (kJ)	282.53	280.12	284.26	846.91
	Visited Nodes	49	53	53	**155**
	Total Rewards	15.12	24.72	23.40	**63.24**
	Process Time (s)				**1.1473**
HIPP	Path Length (m)	2266.11	2253.35	2244.25	6763.71
	Turning Angles (°)	1357.51	1466.44	1450.19	4274.14
	Completion Time (s)	1155.68	1151.12	1146.29	3453.09
	Energy Consumption (kJ)	287.26	287.66	286.32	861.24
	Visited Nodes	51	50	49	150
	Total Rewards	17.60	16.90	18.50	53.00
	Process Time (s)				1.5072

Table 5.4 Performance comparison of VAP & EA-MIPP and HIPP for Mission 4 (multi-UAV case with 3 UAVs). $T_{\text{bud}} = 1350$ s, $E_{\text{bud}} = 324$ KJ.

Algorithm	**Metrics**	**UAV 1**	**UAV 2**	**UAV 3**	**Total**
EA-MIPP	Path Length (m)	2312.45	2321.54	2326.57	6960.56
	Turning Angles (°)	3133.34	2924.25	3021.05	9078.64
	Completion Time (s)	1208.44	1209.51	1213.63	3631.58
	Energy Consumption (KJ)	323.38	320.82	323.08	967.28
	Visited Nodes	59	59	59	**177**
	Total Rewards	23.20	26.22	25.90	**75.32**
	Process Time (s)				**1.2843**
HIPP	Path Length (m)	2518.96	2531.55	2532.66	7583.17
	Turning Angles (°)	1445.66	1575.33	1674.37	4695.36
	Completion Time (s)	1283.57	1292.03	1294.23	3869.83
	Energy Consumption (KJ)	318.22	321.93	323.77	963.92
	Visited Nodes	56	55	55	166
	Total Rewards	18.40	18.80	21.30	58.50
	Process Time (s)				1.6829

Table 5.5 Performance comparison of VAP & EA-MIPP and HIPP for Mission 5 (multi-UAV case with 3 UAVs). $T_{\text{bud}} = 1500$ s, $E_{\text{bud}} = 360$ KJ.

Algorithm	**Metrics**	**UAV 1**	**UAV 2**	**UAV 3**	**Total**
EA-MIPP	Path Length (m)	2540.48	2583.90	2629.04	7753.42
	Turning Angles (°)	3120.95	3365.92	3093.38	9580.25
	Completion Time (s)	1322.25	1348.05	1366.08	4036.38
	Energy Consumption (KJ)	349.70	359.00	359.54	1068.24
	Visited Nodes	65	65	63	**193**
	Total Rewards	26.82	28.02	28.45	**83.29**
	Process Time (s)				**1.3420**
HIPP	Path Length (m)	2812.73	2804.68	2806.71	8424.12
	Turning Angles (°)	1716.84	1725.52	1911.35	5353.71
	Completion Time (s)	1434.98	1431.10	1435.21	4301.29
	Energy Consumption (KJ)	357.10	356.32	359.77	1073.19
	Visited Nodes	62	61	60	183
	Total Rewards	20.50	21.30	23.75	65.55
	Process Time (s)				1.7241

Across all missions, VAP with EA-MIPP consistently outperformed VAP with HIPP in reward collection and node coverage. Specifically, our method achieved 11–27% higher total rewards, with the advantage becoming more pronounced as mission budgets increased. For instance, in Mission 5, VAP with EA-MIPP visited 193 nodes compared to 183 by VAP with HIPP, yielding a total reward of 83.29 versus 65.55.

While EA-MIPP produced longer paths and higher cumulative turning angles, these additional maneuvers translated into greater mission utility without violating the dual constraints. Both methods respected time and energy limits, but EA-MIPP consistently required less computation time, reducing process time by 20–30% compared to HIPP. This demonstrates that the added optimization in EA-MIPP incurs no computational overhead.

Figures 5.5–5.8 represents a sample of the planned paths by EA-MIPP and HIPP in missions 3 and 5.

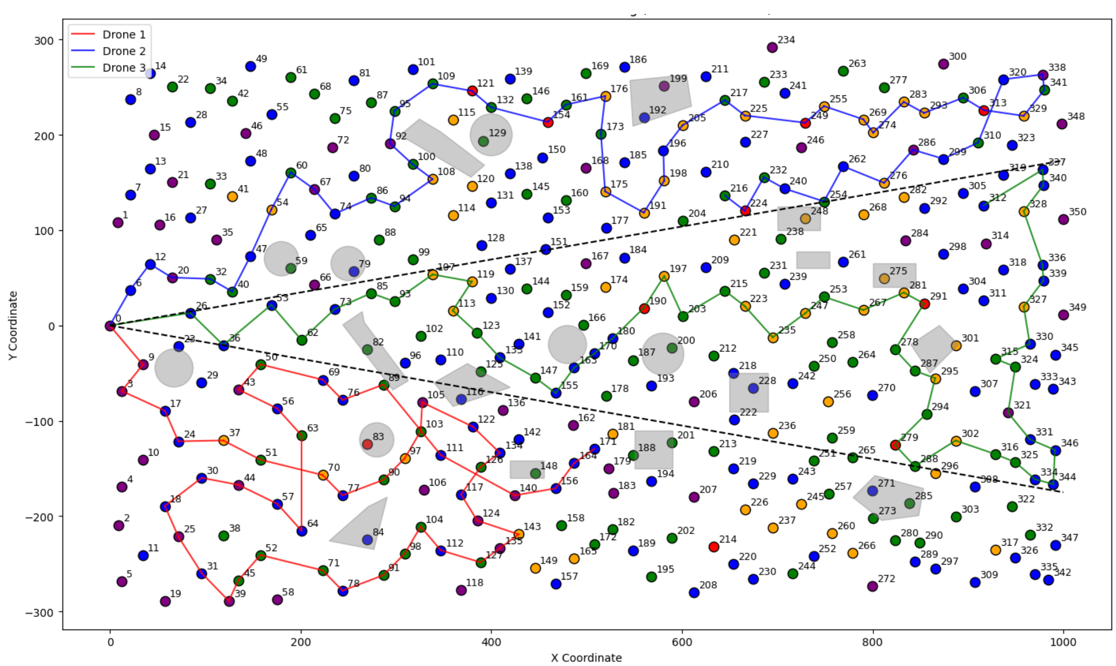


Figure 5.5 EA-MIPP in a 3-UAVs Mission 3.

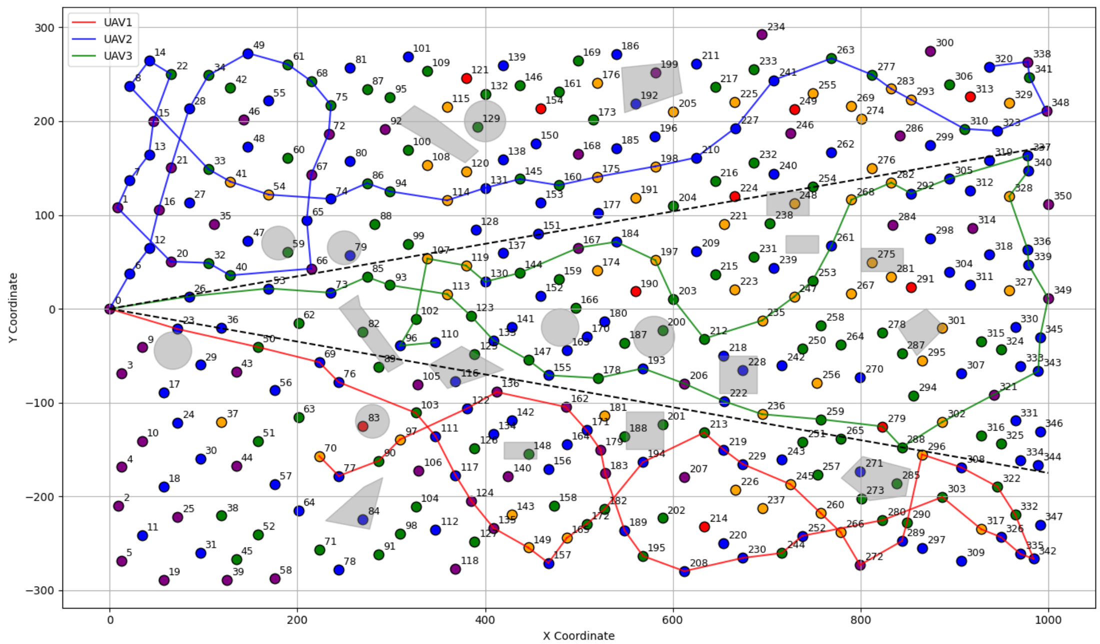


Figure 5.6 HIPP in a 3-UAVs Mission 3.

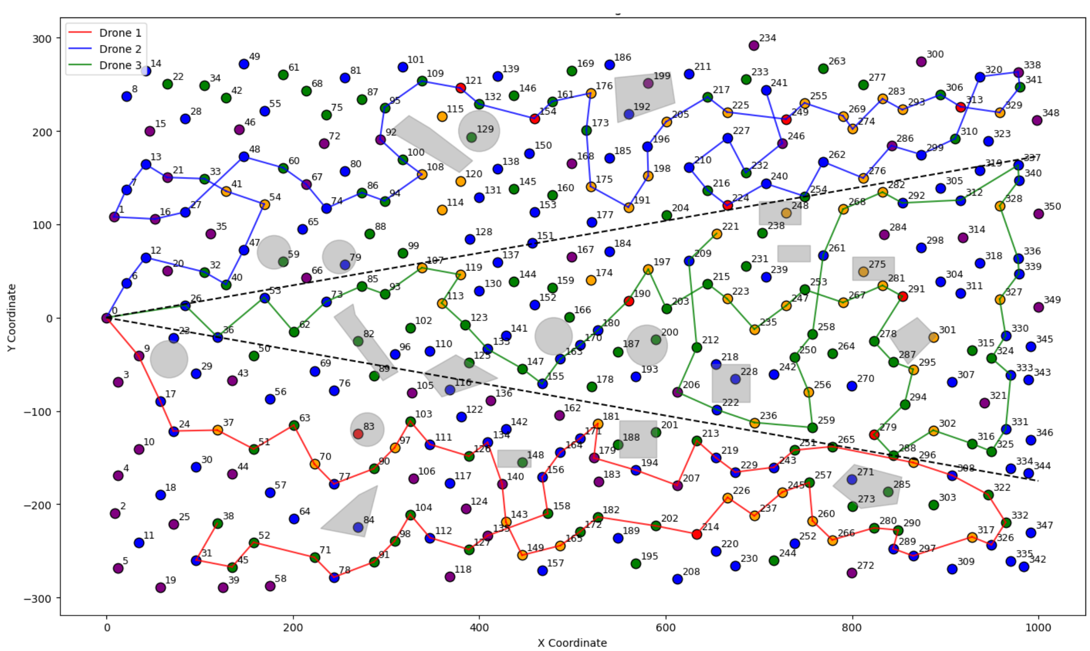


Figure 5.7 EA-MIPP in a 3-UAVs Mission 5.

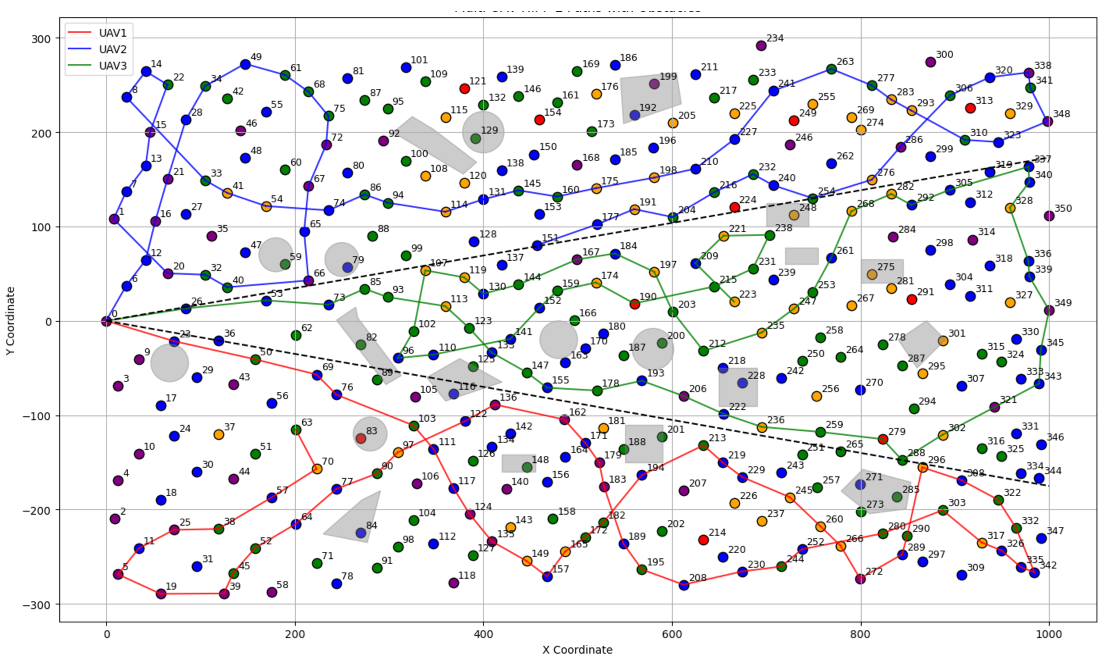


Figure 5.8 HIPP in a 3-UAVs Mission 5.

The evaluation results shows that VAP with EA-MIPP achieves higher rewards, improved node coverage, and shorter planning times than VAP with HIPP while fully respecting dual time-energy constraints. The gains scale with mission size, confirming the robustness of the proposed framework in increasingly demanding multi-UAV operations. This underlines the importance of combining energy-aware planning with VAP-based spatial partitioning for effective coordination of multiple UAVs.

5.5 SUMMARY

In this work, we presented **VAP & EA-MIPP**, a cooperative framework that extends our previous single-UAV algorithm EA-IPP2n to multi-UAV missions. The proposed approach combines V-shape Area Partitioning with Energy-Aware Multi-UAV Informative Path Planning to address the challenges of redundant coverage, unbalanced workloads, and strict mission constraints. VAP ensures fair and non-overlapping partition allocation of Regions of Interest, while EA-MIPP inherits the two-step lookahead mechanism and multi-factor cost model of EA-IPP2n, it extends the framework to cooperative multi-UAV missions by integrating V-shape partitioning and explicitly handling 2D static obstacles and NFZs. This enables safe, efficient, and balanced trajectory planning under strict time and energy constraints.

Simulation results across five missions confirmed that VAP & EA-MIPP outperforms the baseline HIPP in reward collection, node coverage, and computational efficiency, all while respecting mission constraints. The improvements scaled with mission size, underscoring the robustness and scalability of the proposed framework.

CHAPTER VI

EMERGENCY SAFE LANDING SYSTEM (ESLS)

6.1 OVERVIEW

During any mission UAV may face unpredictable circumstances such as communication loss, GPS failure, or other malfunctions, necessitating automated emergency landings [285]. Selecting a safe landing zone is critical.

Emergency landing in GPS-denied or unknown terrains remains challenging. Existing methods often rely on controlled environments with predefined landing pads or fiducial markers [12, 13, 14], static maps [286, 287], or external detection systems [288, 289, 290], limiting their applicability in dynamic or unknown settings. Furthermore, many do not consider human presence or prioritize landing near survivors in search and rescue (SAR) scenarios. Although Safe Landing Zone (SLZ) detection by Tovanche-Picon et al. [291] shows promise in urban crowds, it struggles in dense vegetation or mixed obstacle environments.

Our proposed Emergency Safe Landing System (ESLS) addresses these challenges. It enables real-time detection and avoidance of both static and dynamic obstacles, thereby improving safety during the final phase of the landing. To accurately detect suitable landing zone, ESLS integrates YOLO-v5 [292] with a binary occupancy map, and employs an RGB-D SLAM framework (RTAB-MAP [156]) for 3D environmental mapping and real-time localization. RTAB-MAP leverages visual-inertial odometry (VIO) by fusing data from the onboard IMU and RGB-D camera, ensuring accurate position estimation throughout the landing process. This setup allows the UAV to safely navigate and land in GPS-denied, unknown, and dynamically changing environments such as disaster zones, forests, and dense urban areas. ESLS operates in two distinct modes to enable safe landing zone detection:

- Emergency Safe Landing Zone Detection (ESLZD) mode: the UAV scans its surroundings to identify an obstacle-free landing zone (Section 6.3.1).
- ESLZD-SAR mode: the UAV prioritizes landing near detected survivor by analyzing YOLOv5 bounding boxes (Section 6.3.2).

Once a safe landing zone is detected, the UAV initiates the descent path. RTAB-Map leverages 3D point cloud data and VIO-based odometry to continuously adjust the descent path in real-time, allowing the UAV to autonomously avoid dynamic and static obstacles, minimize obstacle avoidance path energy cost, and land safely without relying on a predefined landing pad.

6.2 BACKGROUND

Various systems have been developed to detect safe landing zones and avoid obstacles using different sensing and processing techniques. Obstacles can enter the UAV's flight path and cause damage or failure to land. These can be either static (e.g., trees, buildings) or dynamic (e.g., other UAVs, birds) [53, 42], with dynamic ones posing greater challenges due to their unpredictable movement and speed [2].

Different prior works have addressed autonomous landing through vision-based systems, often relying on predefined markers or structured environments. Marker-based strategies, such as those by Guo et al.[12] and Wang et al.[286], use AprilTags or ArUco markers with VIO and gradient-based planners for precise landings on static or moving vehicles. These systems achieve high accuracy under controlled conditions but are limited by their dependence on structured markers and known geometries.

Rodriguez-Ramos et al.[288] apply deep reinforcement learning to guide UAVs toward moving robots, while Falanga et al.[289] use multi-modal fusion with LiDAR and vision for real-time pose estimation. Keipour et al. [14] leverage visual servoing to land on vehicles marked with X-shaped pads. While these approaches improve robustness to motion, they rely on external obstacle avoidance algorithms and often lack generalized obstacle modeling.

Semantic understanding has also been explored for autonomous landing. Guerin et al.[287] use semantic segmentation and Bayesian reasoning to identify landing sites in urban areas following SORA guidelines, though assuming static scenes without dynamic obstacle avoidance.

Tullu et al.[290] and Lee et al.[293] apply optical flow, LiDAR, and laser guidance in GPS-denied environments, but face challenges in energy use and computational load. Tovanche-Picon et al.[291] introduce a deep learning framework that uses crowd density maps and Kalman filtering to detect people-free zones in urban settings. While effective with dynamic pedestrians, it does not address aerial obstacles and lacks validation in fully autonomous or safety-critical missions.

In contrast, our proposed ESLS system is designed for real-time onboard deployment on both general and SAR-specific missions. It supports safe landings in dynamic environments with static and dynamic obstacles including ariel obstacles. Table 6.1 provides a summary of representative approaches based on their sensing modes, landing strategies, and obstacle avoidance capabilities.

Table 6.1 Comparison of Safe Landing Zone Detection Methods

Approach	Sensing & Processing	Landing Zone Detection	Avoided Obstacle Types
Marker-Based Landing			
Guo et al. [12]	GPS, Vision Sensors	AprilTag-based landing on moving platform	Static (trees, buildings)
Saavedra-Ruiz et al. [13]	Monocular Camera, IBVS, Kalman Filter	Tracks predefined landing pad	N/A
Keipour et al. [14]	Monocular Camera, Visual Servoing	Detects X-marked pad on moving vehicle	Static & Dynamic (DJI system)
Wang et al. [286]	ArUco markers, Gradient Planner, VIO, Depth Camera	ArUco-based pose estimation	Static (EGO Planner [294])
Semantic Urban-Focused			
Guerin et al. [287]	Semantic Segmentation, Bayesian Monitoring	Urban terrain classification	Urban (buildings, roads)
Learning/Vision-Based			
Rodriguez et al. [288]	Deep Reinforcement Learning	Lands on a moving robot	Static & Dynamic (external system [295])
Falanga et al. [289]	LiDAR + Vision Fusion	Tag-based landing with motion estimation	Static & Dynamic (external system [296])
Tullu et al. [290]	LiDAR, 2-axis gimbal	Optical flow-based tracking	Static & Dynamic
Tovanche-Picon et al. [291]	RGB Camera, DNN, Kalman Filter	Crowd-aware detection using occupancy map	Static & Dynamic (including people)
Lee et al. [293]	Optical Flow, Laser Guidance	Optical flow + laser terrain analysis	Static terrain
Proposed System			
ESLS	RGB-D, IMU, YOLO-v5, VIO, RTAB-MAP	ESLZD and ESLZD-SAR using YOLO + occupancy map	Static & Dynamic (trees, vehicles, aerial objects, etc.)

6.3 EMERGENCY SAFE LANDING SYSTEM (ESLS)

The UAV begins its mission by scanning the area searching for a suitable landing zone. Based on mission requirements, it identifies an obstacle-free landing zone using either ESLZD or ESLZD-SAR. After that the UAV transitions over the detected landing zone area and begins its descent. The descent path is dynamically adjusted using both 3D point cloud data and odometry information from RTAB-MAP, enabling real-time obstacle detection, avoidance, and accurate localization. The ESLS system follows a modular structure: YOLOv5 is used during the detection phase, while RTAB-Map and VIO are activated during descent for mapping and navigation. This ensures safe navigation through unpredictable environments by avoiding both static and dynamic obstacles. The ESLS flowchart is illustrated in Figure 6.1 and visualized in Figure 6.2.

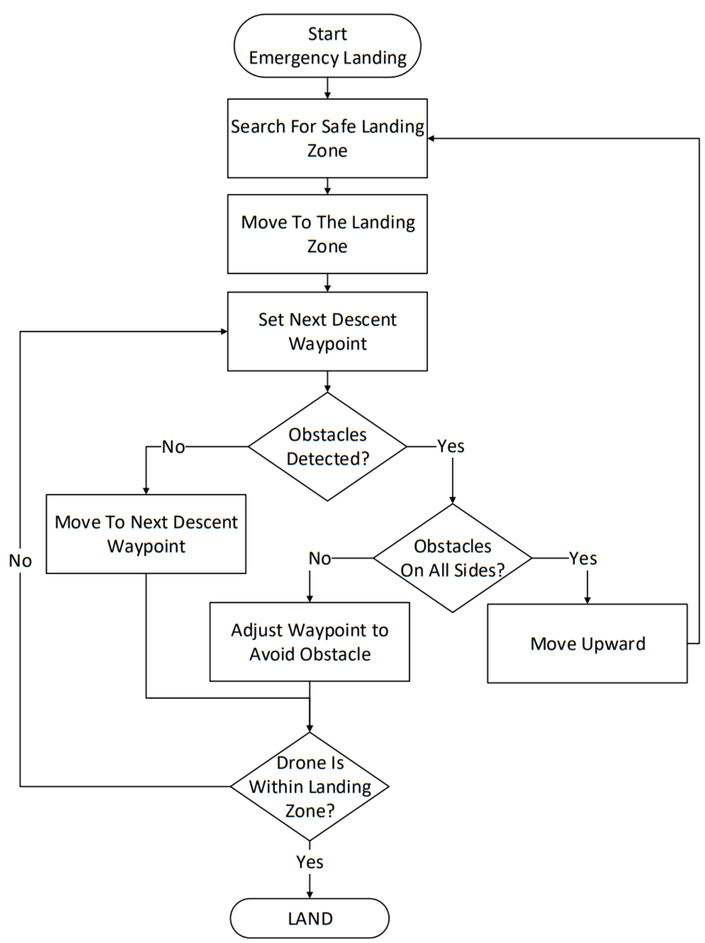
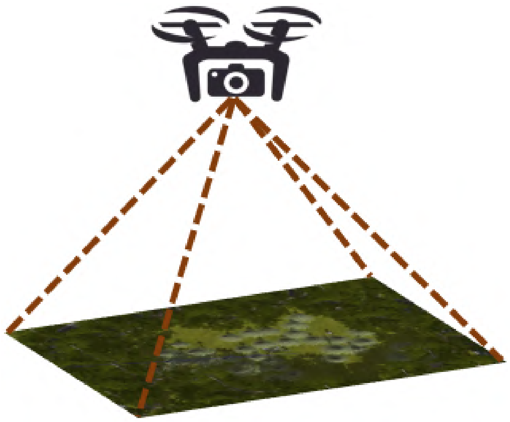
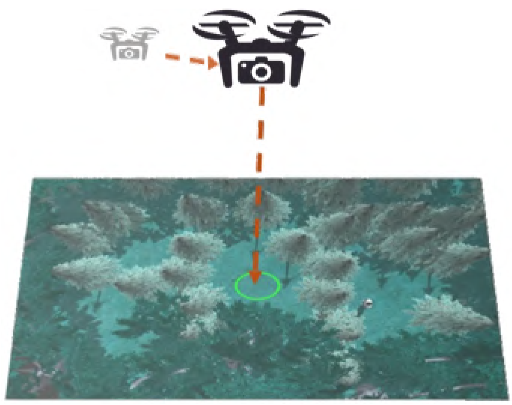


Figure 6.1 Emergency Safe Landing System Flowchart.

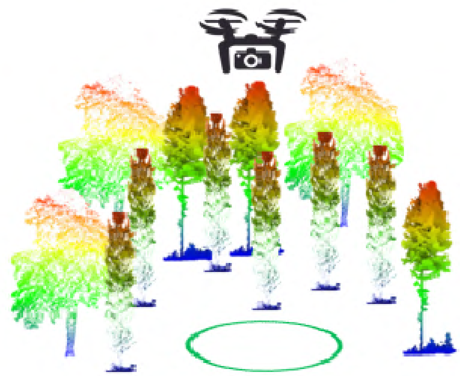


(a) Scanned area

(b) Landing zone detection



(c) Relocating UAV over landing zone



(d) Descent path planning with obstacle detection and avoidance

Figure 6.2 Emergency Safe Landing System Overview.

6.3.1 EMERGENCY SAFE LANDING ZONE DETECTION (ESLZD)

The Emergency Safe Landing Zone Detection (ESLZD) algorithm identifies suitable landing zones in real time by integrating object detection, image-based obstacle masking, color filtering, and geometric safety analysis based on distance transforms. It is designed to operate during emergencies by analyzing the UAV's downward-facing camera feed and detecting obstacle-free zones that satisfy predefined safety constraints.

The ESLZD algorithm 6.1, begins by acquiring an RGB image from the UAV's camera and converting it to grayscale. A binary occupancy map $O \in \{0,1\}^{W \times H}$ is then generated by applying an intensity threshold to highlight obstacle regions. In this map, $O(x,y) = 1$ denotes free space, while $O(x,y) = 0$ denotes obstacles. Additionally, obstacles detected by YOLO-v5 are included by masking the bounding boxes $\mathscr{B}$. Each bounding box $b \in \mathscr{B}$ is marked obstacles in the binary map. The binary map with bounding boxes and selected safe landing zone is illustrated in Figure 6.3.

Figure 6.3 ESLZD binary occupancy map.

To eliminate hazardous terrain such as water surfaces or glossy materials that may appear safe in grayscale, color filtering is applied to remove red- or blue-dominant regions, which are often associated with unsafe or non-solid surfaces.

Once the binary map is generated, the algorithm computes the Euclidean distance Eq.2.3 transform $D(x,y)$ over the map:

$$D(x,y) = \min_{\substack{(x',y') \\ O(x',y')=0}} \sqrt{(x-x')^2 + (y-y')^2}$$

This transformation assigns to each pixel the shortest distance to the nearest obstacle. A square region of interest (ROI), $\mathscr{R} \subset O$, is defined around the center of the image, corresponding to the vertical projection of the UAV's position. The search for a landing zone is restricted to this ROI to ensure efficiency and to prioritize closest landing zone. The corresponding distance transform visualization in Figure 6.4 highlights safety contours using color gradients, where warmer tones represent higher clearance from surrounding obstacles.

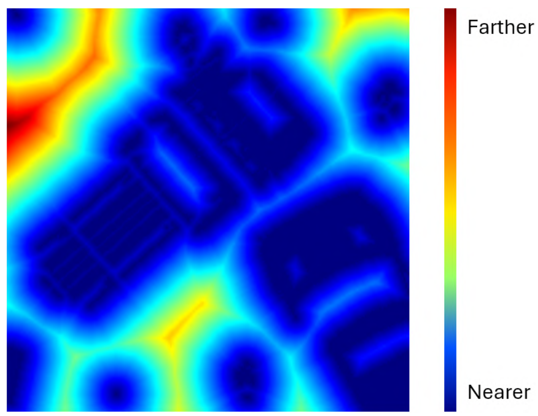


Figure 6.4 Distance transform visualization of ROI. Where warmer colors indicate pixels farther from obstacles.

The goal is to find the point $(x^*, y^*) \in \mathscr{R}$ that is maximally distant from all obstacles:

$$(x^*, y^*) = \arg \max_{(x,y) \in \mathscr{R}} D(x, y)$$

This optimal point must satisfy two constraints to be considered a valid landing zone. First, the available free space radius must meet or exceed a minimum safety threshold $r_{\text{safe}} \geq 3r_{\text{UAV}}$, where r_{UAV} is the UAV's physical radius:

$$D(x^*, y^*) \geq r_{\text{safe}}$$

Second, the point must not lie within any bounding box associated with a detected object:

$$(x^*, y^*) \notin \bigcup_{b \in \mathscr{B}} b$$

If these conditions are met, the point is selected as the landing zone. Otherwise, the process continues until a valid zone is identified. Figure 6.5 shows a real-time snapshot of the ESLZD algorithm's visualized output.

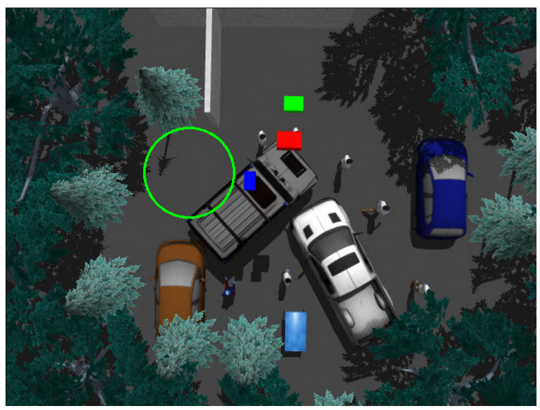


Figure 6.5 ESLZD output: green circle marks landing zone; colored boxes indicate aerial obstacles (e.g., birds, UAVs)

6.3.2 ESLZD-SAR IN THE PRESENCE OF A SURVIVOR

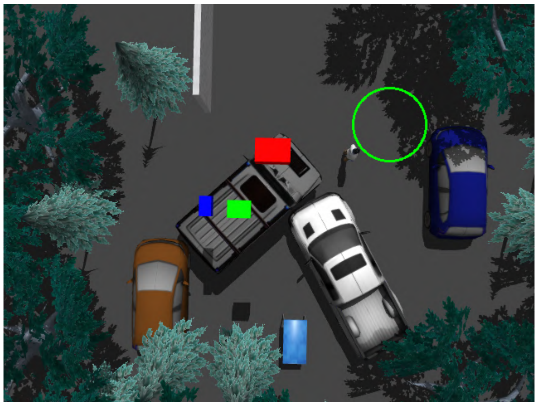
The ESLZD-SAR mode extends the ESLZD algorithm to prioritize landing near a detected survivor for SAR operations. YOLO-v5 identifies the survivor and provides bounding box coordinates, which define a focused Region of Interest (ROI) centered on the survivor.

Input: RGB image, YOLO-v5 bounding boxes
Output: Safe landing zone center coordinates

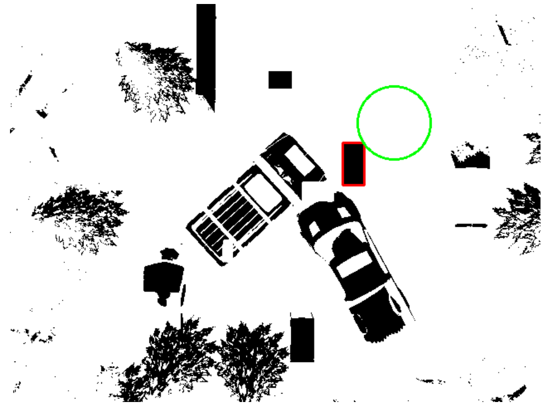
Initialize
    Safe zone radius threshold r_{safe};
    ROI parameters;
while *landing zone not found* **do**
    Convert image to grayscale and threshold to obtain binary occupancy map O;
    Mask all YOLO-detected bounding boxes $\mathscr{B}$ in O;
    Apply color filtering to remove red/blue unsafe regions;
    Define a central ROI $\mathscr{R} \subset O$;
    Compute Euclidean distance transform $D(x,y)$ over $\mathscr{R}$;
    Find $(x^*,y^*) = \arg\max_{(x,y)\in\mathscr{R}} D(x,y)$;
    if $D(x^*,y^*) \geq r_{safe}$ ***and*** $(x^*,y^*) \notin \bigcup_{b\in\mathscr{B}} b$ **then**
        Mark landing zone as found;
        Publish (x^*,y^*) as landing coordinates;
    end
    else
        Continue image processing;
    end
end

Algorithm 6.1: Emergency Safe Landing Zone Detection (ESLZD)

Within this ROI, a binary occupancy map is constructed—excluding obstacles and the survivor bounding box itself—to isolate safe areas for descent. A Euclidean distance transform is then applied to the localized ROI (see Figures 6.6a and 6.6b), identifying the point furthest from any nearby obstacles as the safest potential landing zone. Once a valid landing zone is detected, the UAV proceeds to land near the survivor to initiate communication (via speaker or microphone), deliver emergency supplies, or await rescue teams all while minimizing energy consumption.



(a) ESLZD-SAR output showing proximity to the detected survivor.



(b) Binary occupancy map with YOLOv5-detected survivor highlighted (red box).

Figure 6.6 ESLZD-SAR visual results.

6.3.3 TRANSFORMATION TO REAL-WORLD COORDINATES

COORDINATE CONVERSION

To relocate the UAV to its next position above the landing zone, the coordinates of the detected landing zone in the YOLO image frame must be converted to the real world coordinates relative to the position of the UAV. This transformation accounts for the UAV's altitude and camera's FOV. In Figure 6.7 the YOLO image (rectangle BCDE) represents the area captured by the UAV's downward-facing camera. The image resolution is 640×480 pixels.

The center of the YOLO image frame is denoted as M, which represents the projection of the UAV's position A onto the YOLO image. The center of the detected landing zone (green circle) is denoted by L, and its inverse projection corresponds to the next UAV position N. The

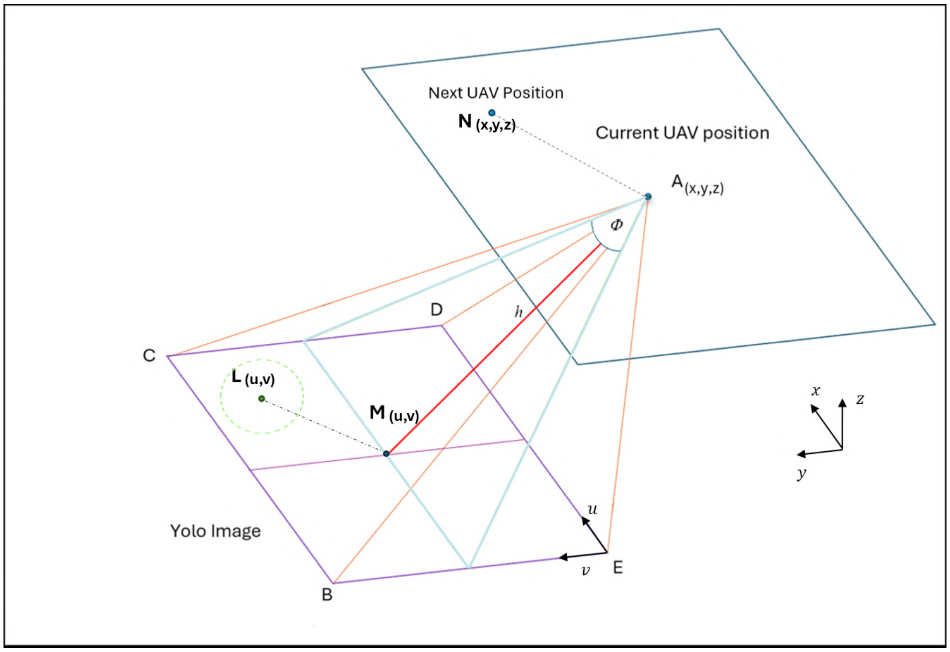


Figure 6.7 YOLO scanned area dimensions in 3D view.

dimensions of the scanned area (BCDE) depend on the camera's FOV and the UAV's altitude h ($h = AM$).

FIELD OF VIEW CALCULATIONS

The horizontal FOV is given $\phi_h = 62^\circ$. Converting it to radians:

$$\phi_h = \frac{62^\circ \cdot \pi}{180} \tag{6.1}$$

The vertical FOV is calculated based on the image aspect ratio:

$$\phi_v = 2 \cdot \arctan\left(\tan\left(\frac{\phi_h}{2}\right) \cdot \frac{h}{w}\right) \tag{6.2}$$

The real-world dimensions of the YOLO image frame at altitude h are:

$$W = 2 \cdot h \cdot \tan\left(\frac{\phi_h}{2}\right), \qquad H = 2 \cdot h \cdot \tan\left(\frac{\phi_v}{2}\right) \tag{6.3}$$

PIXEL-TO-WORLD MAPPING

Each pixel in the YOLO image corresponds to a real-world coordinate displacement relative to the center of the frame (the UAV's projected position). The image center coordinates $M(u_M, v_M)$ are:

$$u_M = \frac{w}{2}, \qquad v_M = \frac{h}{2} \tag{6.4}$$

For a detected target at pixel position $L(u_L, v_L)$, the displacement in pixel coordinates is:

$$\Delta u = u_L - u_M, \qquad \Delta v = v_L - v_M \tag{6.5}$$

These displacements are converted into real-world distances $(\Delta x, \Delta y)$ using the following relationships:

$$\Delta x = \frac{\Delta u}{w} \times W, \quad \Delta y \qquad = \frac{\Delta v}{h} \times H \tag{6.6}$$

Thus, the real-world coordinates of the next UAV position N relative to the UAV's current position A in the XY plane are:

$$x_N = x_A + \Delta x, \qquad y_N = y_A + \Delta y \tag{6.7}$$

This transformation accurately localizes the detected landing zone in real-world coordinates, enabling precise UAV navigation toward the target point.

6.3.4 REAL-TIME DESCENT & OBSTACLE AVOIDANCE

The UAV performs a real-time descent toward the ground using VIO for localization and RTAB-Map's 3D point cloud for obstacle detection. Each point in the cloud is transformed into the UAV's local coordinate frame and evaluated based on its relative position (x, y, z). If a point falls within a predefined safety threshold in any direction, that zone is marked as obstructed. The point cloud is segmented into directional zones relative to the UAV's body frame:

$$\mathscr{D}\mathscr{Z} = \{\text{front}, \text{back}, \text{left}, \text{right}, \text{below}\}.$$

To ensure safe descent, the UAV follows a waypoint-based strategy: at each iteration, it analyzes its surroundings, checks for obstacles within a critical proximity (e.g., 0.5 m), and generates a new waypoint accordingly. If no obstacles are detected, it descends; if all lateral directions are blocked, it ascends slightly to escape the local trap and re-evaluates the environment. Otherwise, it samples candidate waypoints in cardinal and diagonal directions, evaluates each based on its total energy cost, and selects the optimal direction. The next waypoint is computed by offsetting the current position by the direction vector with the lowest energy cost:

$$\vec{d}^* = \arg\min_{\vec{d}_i \in \mathscr{D}} \text{Energy}(\vec{d}_i),$$

The full decision logic for the next waypoint $\vec{w}_{t+1}$ is:

$$\vec{w}_{t+1} = \begin{cases} \vec{p}_t + (0, 0, -\Delta z), & \text{no obstacles} \\ \vec{p}_t + (0, 0, +\Delta z), & \text{fully blocked} \\ \vec{p}_t + \vec{d}^*, & \text{otherwise} \end{cases}$$

where $\vec{d}_i$ represents directional offsets from the UAV's current position. This real-time decision model is executed in every control cycle to ensure safe descent while minimizing energy consumption and maneuvering cost.

If no prior waypoint history is available, the UAV avoids obstacle in any detected obstacle-free direction. The UAV selects the candidate waypoint with the minimum total energy cost based on the energy model proposed in [297] Eq.2.7.

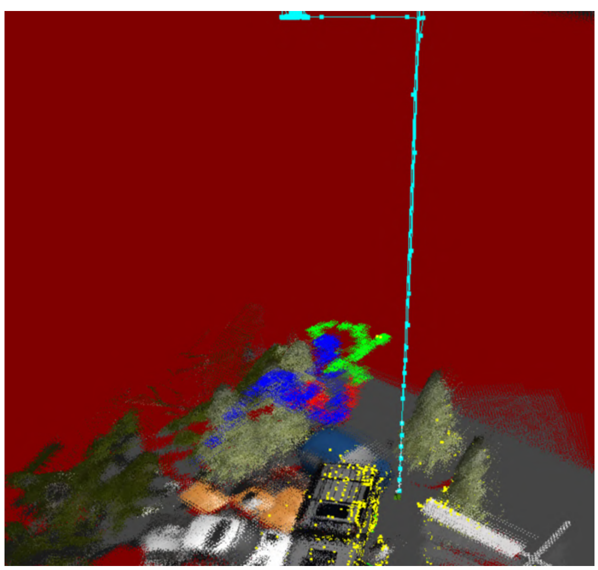


Figure 6.8 UAV descent and obstacle avoidance path (cyan waypoints) computed from RTAB-Map point cloud.

Algorithm 6.2 summarizes the real-time control loop executed during descent.

Input: VIO pose, point cloud data, velocity, IMU yaw rate
Output: Waypoint commands for safe descent

while *UAV has not landed* **do**
    Process point cloud and classify obstacles by direction;
    if *no obstacles detected* **then**
        Descend: $\vec{w}_{t+1} \leftarrow \vec{p}_t + (0, 0, -\Delta z_{\text{descent}})$;
    else
        if *all lateral directions blocked* **then**
            Ascend: $\vec{w}_{t+1} \leftarrow \vec{p}_t + (0, 0, +\Delta z_{\text{ascend}})$;
        else
            if *waypoint history exists* **then**
                Generate candidate waypoints $\vec{d}_i$;
                Compute energy for each $\vec{d}_i$, select $\vec{d}^*$;
                $\vec{w}_{t+1} \leftarrow \vec{p}_t + \vec{d}^*$;
            else
                Move in clear direction using fixed offset;
            end
            Send movement command;
        end
    end
    if *UAV altitude* $< 0.5\,\text{m}$ **then**
        Land;
        break;
    end
end

Algorithm 6.2: Real-Time Descent Path and Obstacle Avoidance

6.4 SIMULATION & EXPERIMENTAL SETUP

6.4.1 EXPERIMENTAL SETUP

The system was tested in the Gazebo simulator with ArduPilot SITL [298, 299] for real-time landing zone detection and obstacle avoidance (**120 simulation**). Simulations ran on Ubuntu 20.04 with ROS Noetic, using an Intel i7-6700HQ CPU, 16 GB RAM, and Nvidia

GTX 965M GPU. The Iris quadcopter model, equipped with an RGB-D camera, was controlled via MAVROS [300] for UAV-simulator communication.

The ESLS system processes RGB-D data at 30 Hz, with the main obstacle-aware descent loop operating at 2 Hz. ROS-based asynchronous callbacks (e.g., point cloud, IMU) allow real-time obstacle updates between control steps.

6.4.2 SIMULATION RESULTS

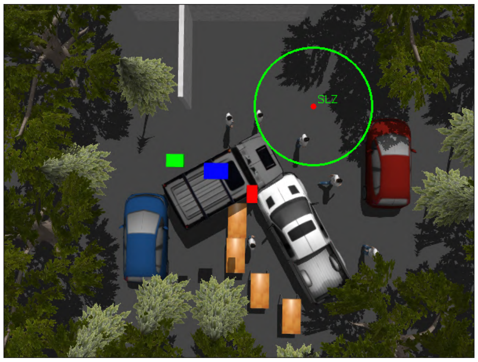
We conducted three sets of evaluations: (1) a comparative analysis against a baseline method, (2) evaluation of ESLZD, and (3) evaluation of ESLZD-SAR. All tests were conducted in two representative environments: a static forest with dense obstacles and a mixed static-dynamic environment containing aerial and ground obstacles.

BASELINE COMPARISON WITH SLZ

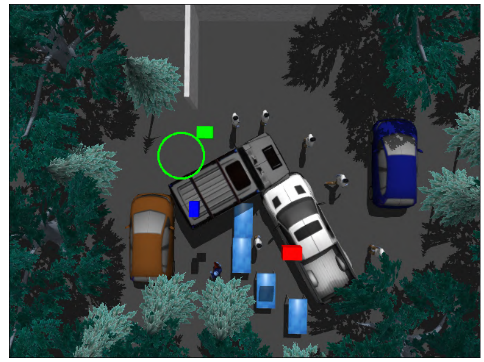
Table 6.2 Accuracy and average processing time (Avg PT) of safe landing zone detection using SLZ [291] and the proposed ESLZD under varying obstacle densities in dynamic and forest environments.

Scenario	Method	Metric	Low	Moderate	High
Dynamic	SLZ	Accuracy	96.66%	66.66%	46.67%
		Avg PT (s)	0.115	0.124	0.135
	ESLZD	Accuracy	100%	100%	96.66%
		Avg PT (s)	0.065	0.085	0.099
Forest	SLZ	Accuracy	90.00%	40.00%	16.67%
		Avg PT (s)	0.118	0.133	0.148
	ESLZD	Accuracy	100%	96.66%	90.00%
		Avg PT (s)	0.044	0.057	0.069

Under moderate density (Figs. 6.9a–b), both detect valid zones, but ESLZD detected a closer, safer zone. At high density (Figs. 6.9c–d), SLZ fails to find any zone while ESLZD detected a safe area.



(a) SLZ detect valid zones.



(b) ESLZD detected a closer, safer location.

(c) SLZ failed to detect landing zone.

(d) ESLZD detected a safe area.

Figure 6.9 Comparison of SLZ [291] and the proposed ESLZD in two environments. (a–b) Dynamic scenario (moderate density), (c–d) Forest scenario (high density).

EVALUATION OF ESLS SYSTEM IN ESLZD AND ESLZD-SAR MODES

The ESLS system was evaluated in both ESLZD and ESLZD-SAR modes across forest and dynamic environments using the following key metrics:

- *Landing Zone Detection Rate*, which represents the percentage of successful detections of a viable landing zone in ESLZD or near the survivor in ESLZD-SAR.

- *Landing Success Rate*, defined as the percentage of simulations in which the UAV safely landed without colliding with any static or dynamic obstacles, while maintaining a safe distance from the survivor in ESLZD-SAR mode.

- *Average Path Length*, representing the total distance traveled during descent, providing insight into the efficiency of the landing trajectory; and, for ESLZD-SAR.

- *Average Safety Distance*, which quantifies the horizontal distance between the UAV and the survivor after landing (with a minimum required distance of 2 m).

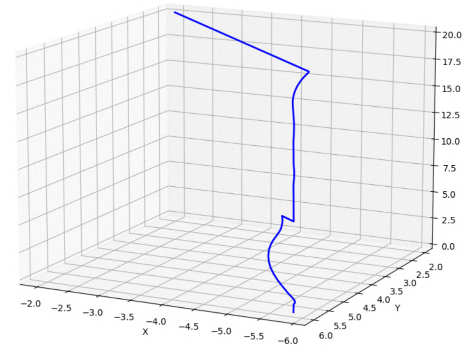
A failure case is defined as any situation where the UAV is unable to detect a safe landing zone in ESLZD, or near the survivor in ESLZD-SAR. Failures also occur if the UAV lands closer than the 2 m safety threshold to the survivor in ESLZD-SAR, or if it touches down on unsuitable surfaces, such as the roof of a vehicle, in either mode.

Figures 6.10 , 6.11 and 6.12 shows a sample of detected landing zones and corresponding landing paths for ESLZD and ESLZD-SAR, respectively. Table 6.3 summarizes the performance of ESLS in both modes across simulation trials conducted in the two environments.

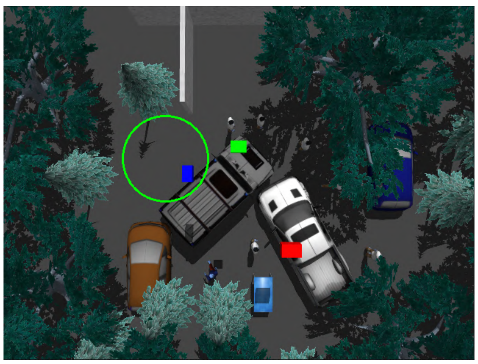
Table 6.3 Performance of ESLS in ESLZD and ESLZD-SAR modes across two scenarios (30 runs per case). Safety distance applies only to ESLZD-SAR.

Metric	**Forest**		**Dynamic**	
	ESLZD	**ESLZD-SAR**	**ESLZD**	**ESLZD-SAR**
Landing Zone Detection (%)	96.66	96.66	96.66	100.00
Landing Success (%)	93.33	96.66	96.66	96.66
Path Length (m) $\pm \sigma$	26.75 ± 1.52	27.32 ± 1.80	24.55 ± 1.73	25.64 ± 2.14
Safety Dist. (m) $\pm \sigma$	N/A	2.42 ± 0.28	N/A	2.96 ± 0.35

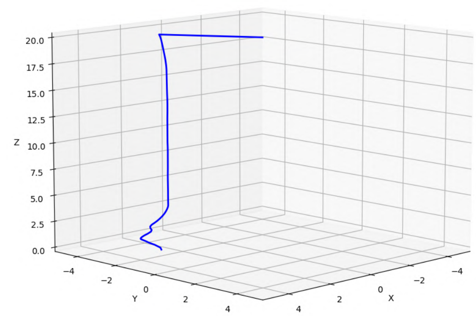
(a) UAV identifies landing zone.



(b) UAV descends to a safe zone.



(c) Landing zone detection.

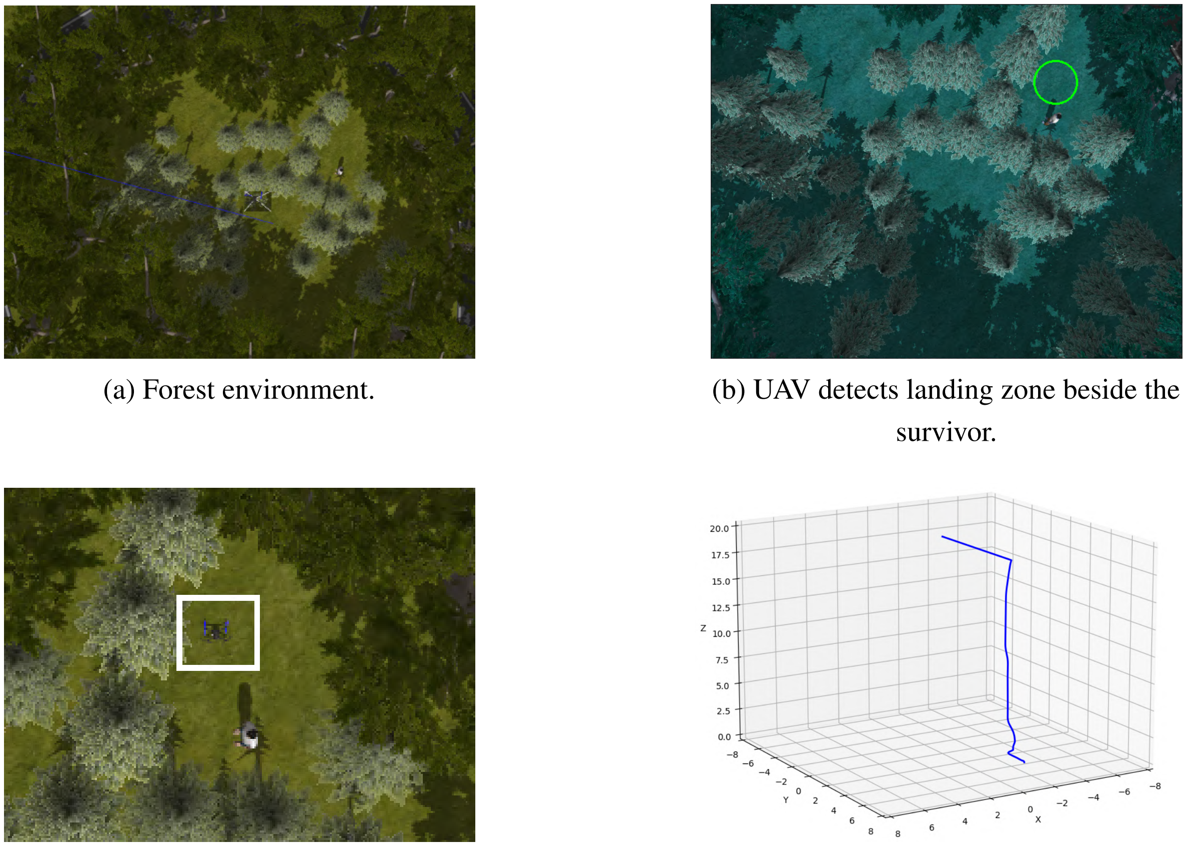


(d) UAV landing path.

Figure 6.10 UAV landing using ESLZD in two scenarios: Forest and Complex dynamic environment.

Results show that in the ESLS with ESLZD mode, the UAV detected safe zones with a success rate exceeding 96% and achieved landing success rates above 93%, successfully avoiding dense trees (Figs. 6.10a–b) and navigating aerial and ground obstacles (Figs. 6.10c–d).

In the ESLZD-SAR mode, the UAV prioritized landing near detected survivors while maintaining safe margins (Figs. 6.11 and 6.12), achieving detection and landing success rates near 97%, with only minor failures occurring when the UAV landed too close to a survivor or when no valid nearby landing zone was available.



(a) Forest environment.

(b) UAV detects landing zone beside the survivor.

(c) UAV lands safely nearby survivor.

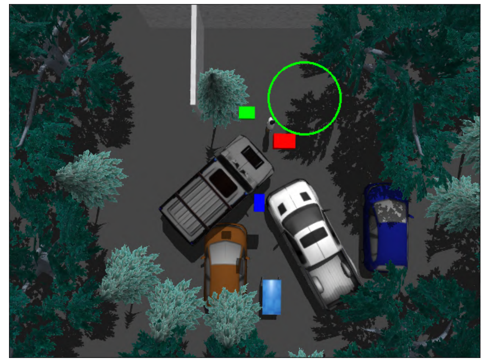
(d) Landing path.

Figure 6.11 UAV landing using ESLZD-SAR in a Forest environment.

6.4.3 LIMITATIONS

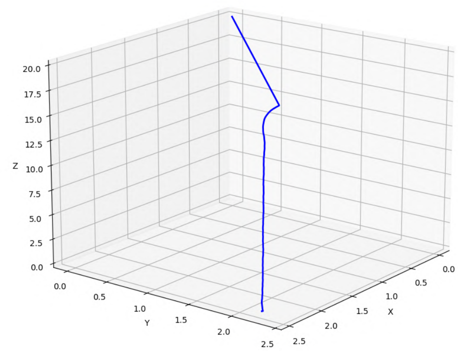
While the ESLS system demonstrates strong performance in both dense and dynamic environments through simulation, several limitations remain. The current implementation relies on an RGB-D camera, which may be affected by low-light conditions or reflective surfaces, potentially impacting landing zone detection accuracy. In the presence of challenging lighting or occlusions, performance may degrade. Future work will explore the integration of thermal or low-light cameras to improve robustness in vision-impaired conditions.

(a) Complex dynamic environment.



(b) ESLZD-SAR detection.

(c) UAV navigates aerial and ground obstacles to land within safe proximity to the survivor.



(d) Landing path.

Figure 6.12 UAV landing using ESLZD-SAR in a Complex dynamic environment.

6.5 SUMMARY

Chapter 6 presented the the Emergency Safe Landing System (ESLS) for UAVs, featuring two landing zone detection modes: ESLZD and ESLZD-SAR. The system effectively identifies safe landing zones in diverse and complex environments, adapting dynamically to obstacles. Compared to the baseline SLZ algorithm, ESLZD consistently found safer and more efficient landing zones. It reduced landing path length and time in dynamic environments and succeeded where SLZ failed in dense forested areas. In ESLZD mode, the UAV navigated safely through aerial and ground obstacles using a real-time descent planner integrating RTAB-Map 3D mapping with VIO for GPS-denied localization. The system continuously evaluates and adapts the descent path to avoid collisions and conserve energy. In SAR mode, ESLZD-SAR detected survivors and landed nearby while maintaining safety constraints, achieving high landing success and proximity metrics in challenging environments. These results demonstrate ESLS's robustness and reliability, making it suitable for both standard and SAR UAV missions.

CHAPTER VII

SYSTEM INTEGRATION AND SIMULATION RESULT

7.1 OVERVIEW

This chapter presents the integration of the pre-mission planner (EA-IPP2n) with the ESLS and presents representative simulation results for a SAR mission in Gazebo. The experiment reproduces a SAR scenario in which the UAV follows a pre-mission informative path, detects a survivor using ESLS in ESLZD-SAR mode, and executes a safe descent and landing near the survivor while avoiding obstacles. Figures 7.1–7.3 present the simulation environment, the pre-mission plan, the ESLS outputs, and the planned path execution.

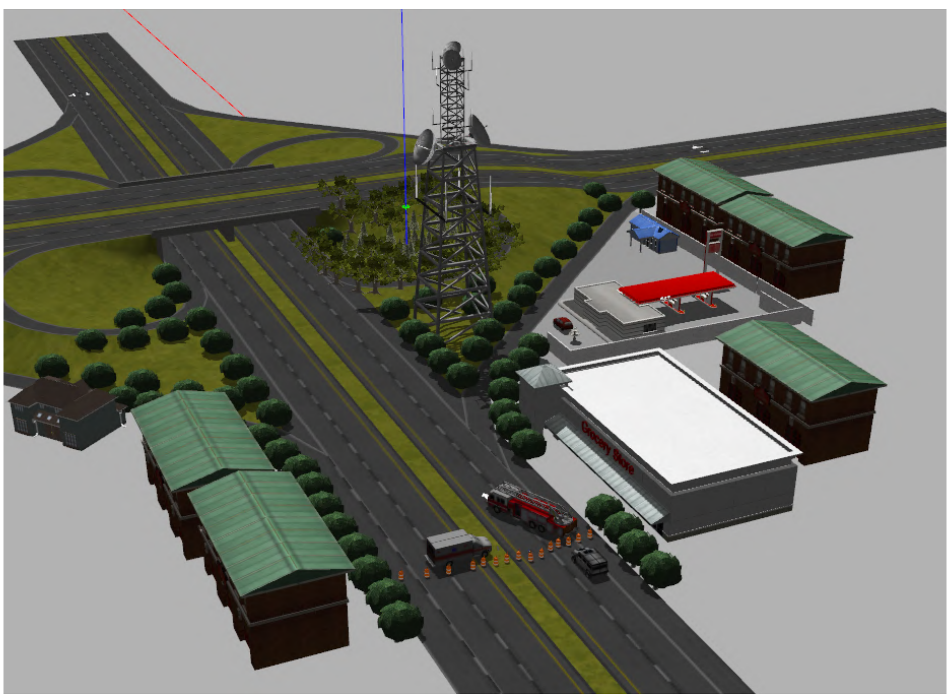


Figure 7.1 Ardupilot - Gazebo environment front view used for pre-mission planning and ESLS testing.

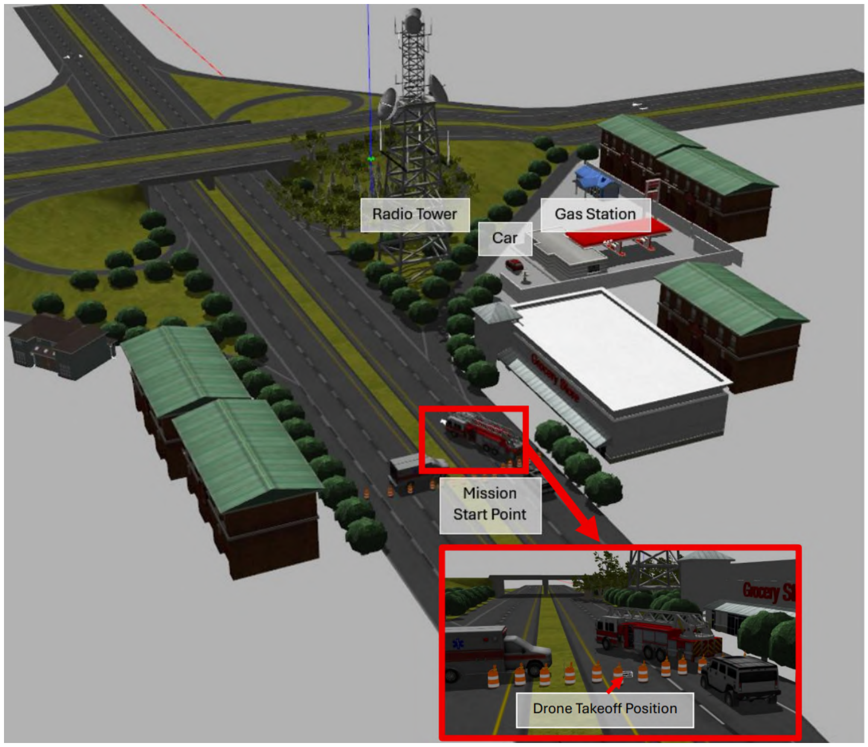


Figure 7.2 AOI front view with initial responder location and terrain.

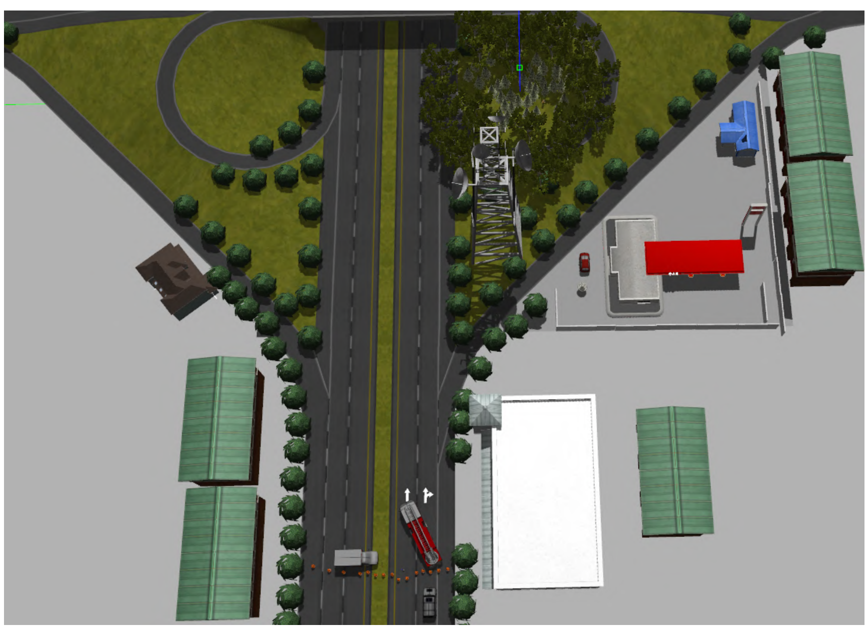


Figure 7.3 AOI — top view.

Figure 7.4 illustrates the overall mission workflow, highlighting the integration between the pre-mission planner (EA-IPP2n) and the ESLS modules.

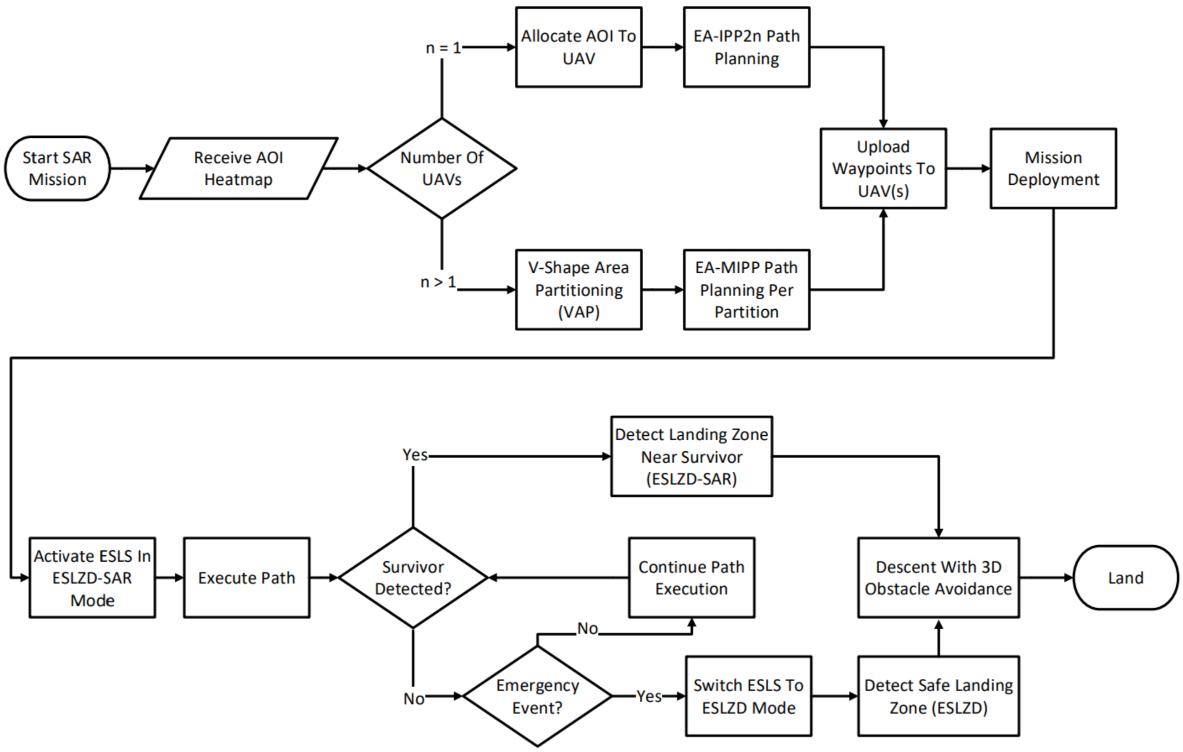


Figure 7.4 Flowchart.

7.2 SIMULATION ENVIRONMENT AND SETUP

A simulation experiment was conducted in Gazebo using an ArduPilot SITL setup [298, 299] on Ubuntu 20.04 with ROS Noetic. The UAV was modeled as an Iris quadrotor equipped with an RGB-D camera and IMU, and controlled via MAVROS for UAV–simulator communication [300]. The perception and localization stack included YOLO-v5, RTAB-Map, and VIO. The experiment was executed on a workstation with an Intel i7-6700HQ CPU, 16 GB RAM, and an Nvidia GTX 965M GPU.

The perception pipeline processed RGB-D data at 30 Hz, while the main obstacle-aware descent loop operated at 2 Hz. ROS-based asynchronous callbacks (e.g., point cloud and IMU updates) provided real-time obstacle information between control steps, enabling safe landing trajectories. This experiment demonstrates the integrated operation of EA-IPP2n and ESLS in a SAR context.

7.3 PRE-MISSION INFORMATIVE PATH PLANNING

The pre-mission planner is an extended implementation of EA-IPP2n that incorporates obstacle awareness, as discussed in Chapter 5. The initial AOI and the spatial priority heatmap are shown in Figures 7.1–7.5.

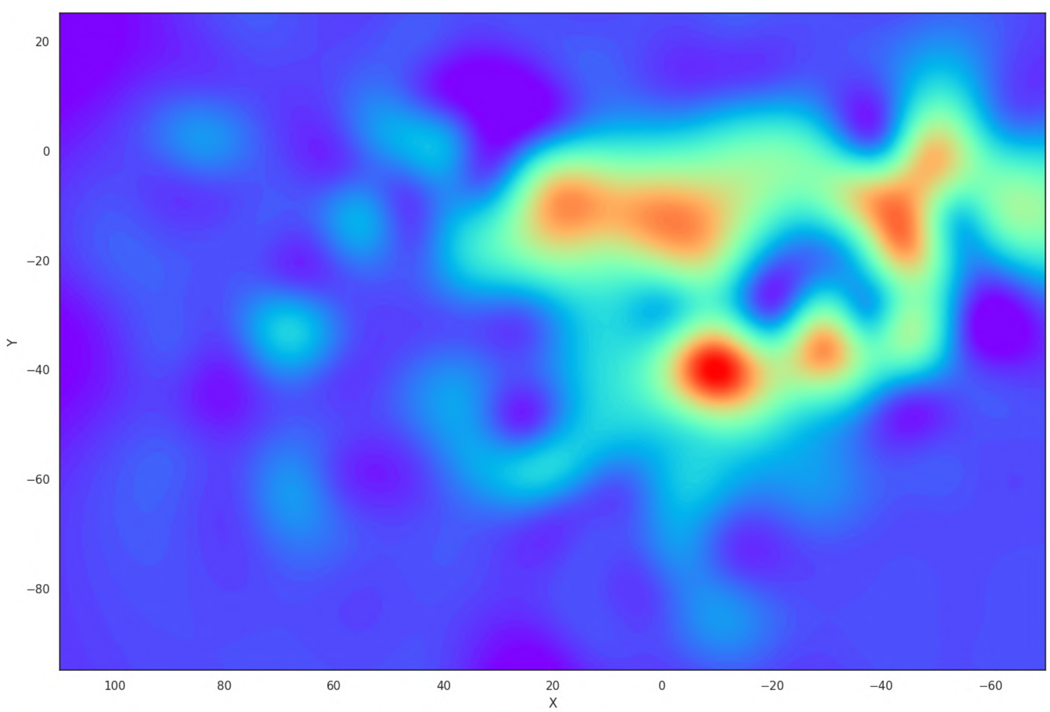


Figure 7.5 Initial heatmap of the AOI representing spatial priority (warmer colors indicate higher priority). The heatmap was the input used to derive node rewards for EA-IPP2n.

In this experiment, the AOI covered 180 × 120 m in the Gazebo world and was discretized into 170 candidate nodes for pre-mission planning. The planner generates a global path over these nodes to maximize the accumulated reward while satisfying both time and energy constraints. The planned trajectory for this experiment is shown in Figure 7.6. At mission start (Figure 7.7), the UAV is positioned at an initial altitude of 40 m and immediately begins following the EA-IPP2n pre-planned path.

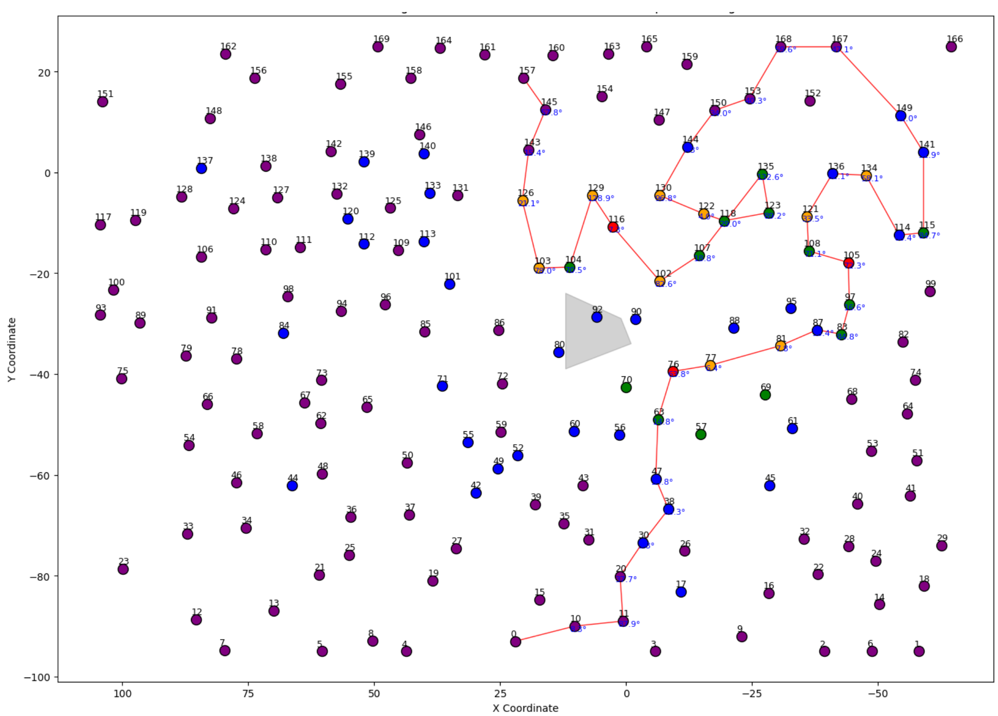


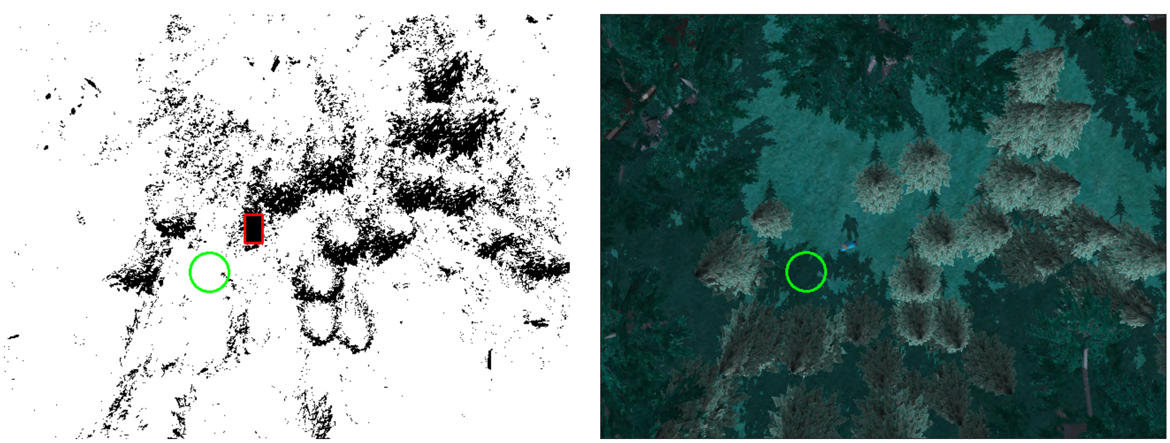
Figure 7.6 Pre-mission path produced by EA-IPP2n with obstacle consideration.

7.4 ESLS EXECUTION AND LANDING

During execution, the UAV follows the planned mission path until YOLO detects a survivor. Once detected, ESLS in ESLZD-SAR mode searches for a safe, obstacle-free landing zone near the detection site to enable landing close to the survivor. Figure 7.8a illustrates the

Figure 7.7 Mission start: the UAV begins its mission at an initial altitude of 40 m and follows the EA-IPP2n trajectory.

binary occupancy representation used by ESLZD-SAR and the YOLO detection; Figure 7.8b shows the ESLZD-SAR outputs.



(a) Binary occupancy map.

(b) ESLZD-SAR detection.

Figure 7.8 ESLS in ESLZD-SAR mode output with detected landing zone (green circle) and YOLO detection of the survivor (red box).

The executed mission path including the local descent and avoidance maneuvers is shown in Figure 7.9. In this run, the planner execution stopped at node 130 when the survivor was detected, and the ESLS descent routine completed the landing sequence.

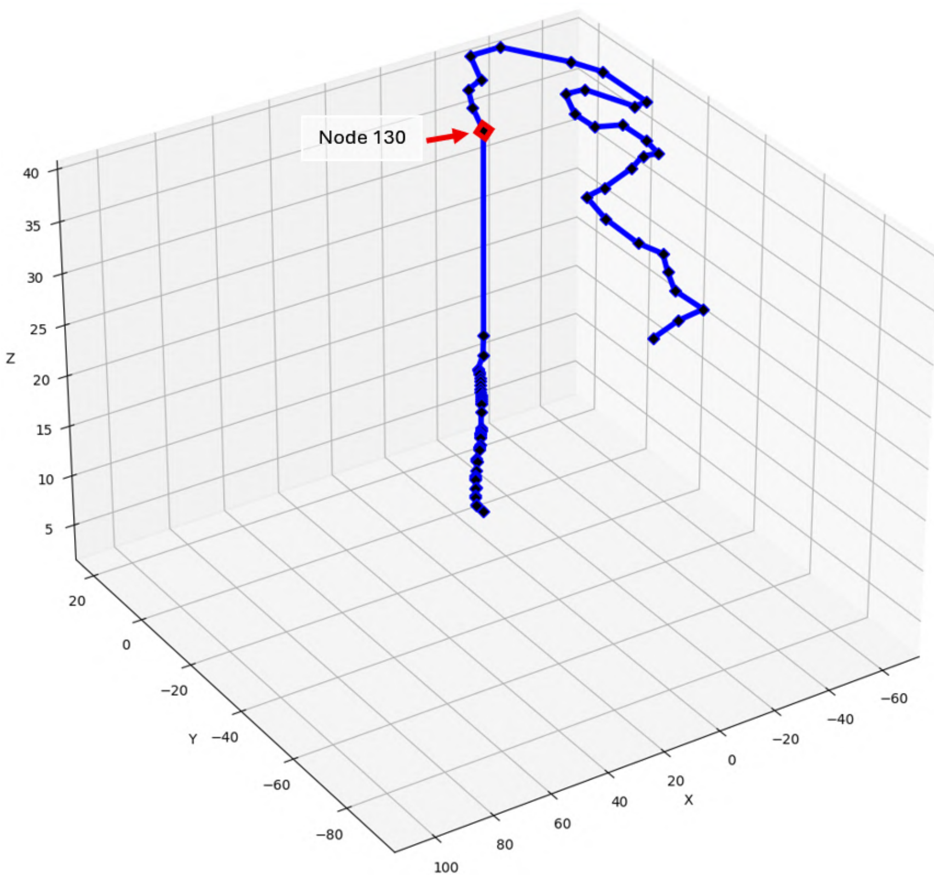


Figure 7.9 Executed UAV mission path from start to landing near the survivor, including EA-IPP2n planning and ESLS-guided descent.

7.5 SIMULATION RESULTS

Table 7.1 reports the primary planning parameters and the mission outcomes recorded for this experiment. The table separates the planner parameters from the outcome metrics to improve readability.

Key observations: The experiment demonstrates the integrated behavior of the pre-mission planner (EA-IPP2n) and ESLS in a SAR mission. Specifically:

- EA-IPP2n generated a reward-aware path that respects both time and energy constraints while exploring high-importance areas (Figures 7.5 and 7.6).
- The UAV detected a survivor at node 130, triggering ESLS in ESLZD-SAR mode to identify a nearby obstacle-free landing zone and plan a safe descent trajectory (Figures 7.8a, 7.8b, and 7.9).
- The executed mission shows the UAV following the EA-IPP2n path and executing a 3D descent guided by ESLS with obstacle-avoidance waypoints, resulting in a safe landing while maintaining a ≈ 2.82 m distance from the survivor.

Table 7.1 Mission parameters and summary metrics for the representative SAR run.

UAV Parameter / Mission Metric	Value
Pre-mission planner settings:	
UAV speed	$v = 1.0$ m/s
Rotational rate (planner)	$\omega = 60.0^\circ$ deg/s
Distance energy coefficient	$\lambda = 0.1164$ KJ/m
Rotational energy coefficient	$\gamma = 0.0173$ KJ/deg
Time budget	$T_{\text{bud}} = 450$ s
Energy budget	$E_{\text{bud}} = 108$ KJ
Pre-Misison Planning (EA-IPP2n)	
Planned path length	411.79 m
Total turning angles	2080.29°
Estimated total completion time	446.46 s
Estimated total energy consumption	83.92 KJ
Nodes visited	43
Total rewards	17.53
Planner computation time	0.1071 s
Executed mission:	
Traversed path length (including descent)	311.64 m
Execution stop condition	Stopped at node 130 (survivor detected)
Safety distance after landing	2.82 m

Notes. The planner provides path length and associated costs based on the graph representation employed by EA-IPP2n; the executed (traversed) path length is expressed in metres as recorded by the simulator after mission termination and includes the local descent and avoidance segments computed by ESLS. The safety distance after landing is measured horizontally from the UAV touchdown location to the survivor's detected position.

7.6 SUMMARY

This chapter demonstrated the integrated operation of the EA-IPP2n pre-mission planner and the ESLS system in a SAR scenario. The UAV successfully executed a global, reward-aware path within energy and time budgets, then switched to vision-based safe landing near a detected survivor. These results highlight the complementary nature of the two modules: strategic resource-aware exploration and adaptive safe landing. Together, they provide a complete mission pipeline that connects the contributions of Chapters 4 - 6.

CHAPTER VIII
CONCLUSION AND FUTURE WORK

8.1 CONCLUSION

UAVs are increasingly used in healthcare, SAR, and emergency response due to their ability to access hazardous or remote environments. Despite their promise, UAVs face persistent challenges in energy efficiency, cooperative path planning, and safe emergency landing.

This thesis addressed these challenges through three core contributions:

- **Energy-Aware Informative Path Planning (EA-IPP2n):** A single-UAV pre-mission planner that balances energy consumption, mission duration, and information gain. By incorporating translational and rotational energy models, a reward-normalized cost function, and a two-hop look-ahead strategy, EA-IPP2n consistently outperformed six baseline methods in both reward collection and efficiency under strict time–energy constraints.
- **VAP-EAMIPP: Cooperative Multi-UAV Planning Framework:** Building on EA-IPP2n, VAP-EAMIPP integrates *V-shape Area Partitioning (VAP)* with an *Energy-Aware Multi-UAV Informative Path Planner* (EA-MIPP). VAP allocates Regions of Interest into balanced angular sectors, minimizing redundant coverage and ensuring fair workload distribution among UAVs. Within each sector, EA-MIPP inherits the two-step lookahead mechanism and multi-factor cost model of EA-IPP2n, while extending it to handle multi-UAV missions with static obstacles and NFZs. Simulation results across five missions showed that VAP-EAMIPP consistently outperformed the baseline Hierarchical

Informative Path Planning (HIPP), achieving higher rewards, greater coverage, and up to 30% lower computational time.

- **Emergency Safe Landing System (ESLS):** A vision-based autonomous landing system that integrates RTAB-Map SLAM with YOLO-v5 object detection for real-time obstacle-aware mapping and safe zone detection. ESLS introduced two operational modes: (i) ESLZD, which autonomously identifies the safest available landing site, and (ii) ESLZD-SAR, which prioritizes proximity to detected survivors. Extensive simulations in Gazebo with ArduPilot demonstrated landing zone detection success rates of up to 98% and safe landing success rates of up to 96%, highlighting robustness in cluttered and GPS-denied environments.

Beyond the individual contributions, Chapter 7 highlighted their integration in a realistic SAR scenario, showing how the pre-mission planner (EA-IPP2n) and emergency landing system (ESLS) interact to provide both global efficiency and local safety. This integration confirms the framework's readiness for real-world UAV deployments.

8.2 FUTURE WORK

While the developed frameworks demonstrated strong performance in simulations, several research directions remain open:

- **Real-World Deployment:** Transition from simulation to physical UAV testing under varying terrains, weather conditions, and sensor noise. Evaluate robustness in GPS-denied and communication-limited environments.

- **Heterogeneous Multi-Agent Coordination:** Extend VAP-EAMIPP to heterogeneous multi-robot teams (e.g., UAV–UGV–USV cooperation) for cross-domain SAR missions with joint aerial, ground, and surface planning.

- **Adaptive and Online Planning:** Integrate dynamic environmental effects such as wind fields and moving obstacles. Develop online re-planning strategies to update paths in real-time, enhancing resilience to unexpected changes.

By addressing these directions, UAVs can evolve into bridging the gap between simulation-based research and real-world deployment.

CHAPTER IX

LIST OF PUBLICATIONS

9.1 HIGHLIGHTS

This doctoral research has been carried out in multiple phases, each addressing specific challenges and resulting in scientific publications. Dissemination of results began as early as the literature review stage, and continued throughout the PhD period until the submission of this thesis.

9.1.1 PEER-REVIEWED JOURNALS

- [1] **Merei, A.**, Mcheick, H., & Ghaddar, A. (2023). Survey on Path Planning for UAVs in Healthcare Missions. *Journal of Medical Systems*, 47, 79. https://doi.org/10.1007/s10916-023-01972-x

- [2] **Merei, A.**, Mcheick, H., Ghaddar, A., & Rebaine, D. (2025). A Survey on Obstacle Detection and Avoidance Methods for UAVs. *Drones*, 9(3), 203. https://doi.org/10.3390/drones9030203

- **Merei, A.**, Mcheick, H., Ghaddar, A., & Beltrame, G. (2025). EA-IPP2n: Energy-Aware Informative Path Planning Algorithm for UAVs. Submitted to *IEEE Access*.

- **Merei, A.**, Mcheick, H., & Ghaddar, A. (2025). VAP & EA-MIPP: V-Shape Area Partitioning for Energy-Aware Multi-UAV Informative Path Planning Missions. To be submitted as an extension of EA-IPP2n.

9.1.2 PEER-REVIEWED CONFERENCE PROCEEDINGS

- **Merei, A.**, Mcheick, H., Ghaddar, A., & Beltrame, G. (2025). ESLS: A Vision-Based Emergency Safe Landing System for UAVs. Accepted for publication in *Procedia Computer Science* (Elsevier). To appear in the Proceedings of the 15th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-25), October 2025.

9.1.3 WORKSHOP & REPORT

In addition to the works directly related to this PhD research, the following contributions were produced during my research internship at the MIST Laboratory, Polytechnique Montréal:

- **Merei, A.**, Mcheick, H., Ghaddar, A., & Beltrame, G. (2025). ESLS: A Vision-Based Emergency Safe Landing System for UAVs. Poster presented at *COLLOQUE REPARTI 2025*, Université Laval, Québec, Canada, May 12, 2025.

- **Merei, A.** (2024). Technical Report: Installation and Simulator Setup for the DJI Matrice Drone in the MIST Laboratory. Polytechnique Montréal, Canada.

REFERENCES

[1] A. Merei, H. Mcheick, and A. Ghaddar, "Survey on path planning for uavs in healthcare missions," *Journal of Medical Systems*, vol. 47, no. 1, p. 79, 2023. [Online]. Available: https://link.springer.com/article/10.1007/s10916-023-01972-x

[2] A. Merei, H. Mcheick, A. Ghaddar, and D. Rebaine, "A survey on obstacle detection and avoidance methods for uavs," *Drones*, vol. 9, no. 3, 2025.

[3] A. Singh, R. Shetty, T. Kale, S. Yadav, and R. Iyer, "Artificial intelligence based drone for healthcare," *International Journal of Recent Advances in Multidisciplinary Topics*, vol. 2, no. 4, pp. 140–144, 2021.

[4] S. Kavirayani, D. S. Uddandapu, A. Papasani, and T. V. Krishna, "Robot for delivery of medicines to patients using artificial intelligence in health care," in *2020 IEEE Bangalore Humanitarian Technology Conference (B-HTC)*. IEEE, 2020, pp. 1–4.

[5] R. Pěnička, J. Faigl, P. Váňa, and M. Saska, "Dubins orienteering problem," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1210–1217, 2017.

[6] Pěnička, Robert, J. Faigl, P. Váňa, and M. Saska, "Dubins orienteering problem with neighborhoods," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2017, pp. 1555–1562.

[7] G. A. Di Caro and A. W. Z. Yousaf, "Multi-robot informative path planning using a leader-follower architecture," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 10 045–10 051.

[8] M. Popović, T. Vidal-Calleja, G. Hitz, J. J. Chung, I. Sa, R. Siegwart, and J. Nieto, "An informative path planning framework for uav-based terrain monitoring," *Autonomous Robots*, vol. 44, no. 6, pp. 889–911, 2020.

[9] A. Stefanopoulou, E. K. Raptis *et al.*, "Improving time and energy efficiency in multi-uav coverage operations by optimizing the uavs' initial positions," *International Journal of Intelligent Robotics and Applications*, vol. 8, pp. 629–647, 2024.

[10] Y. Xiao, Y. Fang, X. Liang, H. Lin, Z. He, and X. Zhao, “Distributed area partitioning for multi-uavs in cooperative monitoring task,” in *2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2018, pp. 1789–1794.

[11] D. Jang, J. Yoo, C. Y. Son, and H. J. Kim, “Fully distributed informative planning for environmental learning with multi-robot systems,” *arXiv preprint arXiv:2112.14433*, 2021.

[12] J. Guo, X. Dong, Y. Gao, D. Li, and Z. Tu, “Simultaneous obstacles avoidance and robust autonomous landing of a uav on a moving vehicle,” *Electronics*, vol. 11, no. 19, p. 3110, 2022.

[13] M. Saavedra-Ruiz, A. M. Pinto-Vargas, and V. Romero-Cano, “Monocular visual autonomous landing system for quadcopter drones using software in the loop,” *IEEE Aerospace and Electronic Systems Magazine*, vol. 37, no. 5, pp. 2–16, 2021.

[14] A. Keipour, G. A. Pereira, R. Bonatti, R. Garg, P. Rastogi, G. Dubey, and S. Scherer, “Visual servoing approach to autonomous uav landing on a moving vehicle,” *Sensors*, vol. 22, no. 17, p. 6549, 2022.

[15] D. Cawthorne and M. H. Frederiksen, “Using the public perception of drones to design for explicability,” in *Proceedings of the 5th International Conference on Robot Ethics and Standards (ICRES)*, Taipei, Taiwan, 2020. [Online]. Available: https://www.updwg.org/wp-content/uploads/2020/09/Design-for-Explicability-Cawthorne-Frederiksen-2020.pdf

[16] K. Yang, Y. Shi, Y. Zhou, Z. Yang, L. Fu, and W. Chen, “Federated machine learning for intelligent iot via reconfigurable intelligent surface,” *IEEE Network*, vol. 34, no. 5, pp. 16–22, 2020.

[17] A. M. Knoblauch, S. de la Rosa, J. Sherman, C. Blauvelt, C. Matemba, L. Maxim, O. D. Defawe, A. Gueye, J. Robertson, J. McKinney *et al.*, “Bi-directional drones to strengthen healthcare provision: experiences and lessons from madagascar, malawi and senegal,” *BMJ global health*, vol. 4, no. 4, p. e001541, 2019.

[18] M. A. H. Zailani, R. Z. A. R. Sabudin, R. A. Rahman, I. M. Saiboon, A. Ismail, and Z. A. Mahdy, “Drone technology in maternal healthcare in malaysia: A narrative review,” *The Malaysian journal of pathology*, vol. 43, no. 2, pp. 251–259, 2021.

[19] E. A. G Peñaloza, V. A Oliveira, and P. E. Cruvinel, “Using soft sensors as a basis of an innovative architecture for operation planning and quality evaluation in agricultural sprayers,” *Sensors*, vol. 21, no. 4, p. 1269, 2021.

[20] I. G. Gallo, M. Martínez-Corbella, R. Sarro, G. Iovine, J. López-Vinielles, M. Hérnandez, G. Robustelli, R. M. Mateos, and J. C. García-Davalillo, “An integration of uav-based photogrammetry and 3d modelling for rockfall hazard assessment: The cárcavos case in 2018 (spain),” *Remote Sensing*, vol. 13, no. 17, p. 3450, 2021.

[21] O. Alvear, C. T. Calafate, N. R. Zema, E. Natalizio, E. Hernández-Orallo, J.-C. Cano, and P. Manzoni, “A discretized approach to air pollution monitoring using uav-based sensing,” *Mobile Networks and Applications*, vol. 23, no. 6, pp. 1693–1702, 2018.

[22] L. Hong, Y. Wang, Y. Du, X. Chen, and Y. Zheng, “Uav search-and-rescue planning using an adaptive memetic algorithm,” *Frontiers of Information Technology & Electronic Engineering*, vol. 22, no. 11, pp. 1477–1491, 2021.

[23] B. Hiebert, E. Nouvet, V. Jeyabalan, and L. Donelle, “The application of drones in healthcare and health-related services in north america: A scoping review,” *Drones*, vol. 4, no. 3, p. 30, 2020.

[24] M. Eichleay, E. Evens, K. Stankevitz, and C. Parker, “Using the unmanned aerial vehicle delivery decision tool to consider transporting medical supplies via drone,” *Global Health: Science and Practice*, vol. 7, no. 4, pp. 500–506, 2019.

[25] J. Phillips, “Medical unmanned aerial system for organ transplant delivery,” *California University of Pennsylvania*, 2019.

[26] A. Barnawi, P. Chhikara, R. Tekchandani, N. Kumar, and M. Boulares, “A cnn-based scheme for covid-19 detection with emergency services provisions using an optimal path planning,” *Multimedia Systems*, pp. 1–15, 2021.

[27] S. S. Fakhrulddin and S. K. Gharghan, “An elderly first aid system based-fall detection and unmanned aerial vehicle,” in *IOP Conference Series: Materials Science and Engineering*, vol. 745, no. 1. IOP Publishing, 2020, p. 012096.

[28] N. Mukati, N. Namdev, R. Dilip, N. Hemalatha, V. Dhiman, and B. Sahu, "Healthcare assistance to covid-19 patient using internet of things (iot) enabled technologies," *Materials Today: Proceedings*, 2021.

[29] M. Nasajpour, S. Pouriyeh, R. M. Parizi, M. Dorodchi, M. Valero, and H. R. Arabnia, "Internet of things for current covid-19 and future pandemics: An exploratory study," *Journal of healthcare informatics research*, pp. 1–40, 2020.

[30] H. S. Munawar, H. Inam, F. Ullah, S. Qayyum, A. Z. Kouzani, and M. Mahmud, "Towards smart healthcare: Uav-based optimized path planning for delivering covid-19 self-testing kits using cutting edge technologies," *Sustainability*, vol. 13, no. 18, p. 10426, 2021.

[31] R. W. Jones and G. Despotou, "Unmanned aerial systems and healthcare: Possibilities and challenges," in *2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2019, pp. 189–194.

[32] S. Azadi, R. Kazemi, and H. Rezaei Nedamani, "Chapter 10 - trajectory planning of tractor semitrailers," in *Vehicle Dynamics and Control*, T. D. Gillespie, Ed. Elsevier, 2021, pp. 429–478. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780323856591000100

[33] T. M. Cabreira, L. B. Brisolara, and P. R. Ferreira Jr, "Survey on coverage path planning with unmanned aerial vehicles," *Drones*, vol. 3, no. 1, p. 4, 2019.

[34] G. Fevgas, T. Lagkas, V. Argyriou, and P. Sarigiannidis, "Coverage path planning methods focusing on energy efficient and cooperative strategies for unmanned aerial vehicles," *Sensors*, vol. 22, no. 3, 2022.

[35] A. Majeed and S. O. Hwang, "A multi-objective coverage path planning algorithm for uavs to cover spatially distributed regions in urban environments," *Aerospace*, vol. 8, no. 11, p. 343, 2021.

[36] S. Nasr, H. Mekki, and K. Bouallegue, "A multi-scroll chaotic system for a higher coverage path planning of a mobile robot using flatness controller," *Chaos Solitons Fractals*, vol. 118, p. 366–375, 2019.

[37] A. Montanari, F. Kringberg, A. Valentini, C. Mascolo, and A. Prorok, “Surveying areas in developing regions through context aware drone mobility,” in *Proceedings of the 4th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, ser. DroNet’18. New York, NY, USA: ACM, 2018, pp. 27–32. [Online]. Available: http://doi.acm.org/10.1145/3213526.3213532

[38] F. Samaniego, J. Sanchis, S. García-Nieto, and R. Simarro, “Recursive rewarding modified adaptive cell decomposition (rr-macd): a dynamic path planning algorithm for uavs,” *Electronics*, vol. 8, no. 3, p. 306, 2019.

[39] M. Coombes, W.-H. Chen, and C. Liu, “Boustrophedon coverage path planning for uav aerial surveys in wind,” in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2017, pp. 1563–1571.

[40] V. M. Samuel, O. M. Shehata, and E.-S. I. Morgan, “Chaos generation for multi-robot 3d-volume coverage maximization,” in *Proceedings of the 4th International Conference on Control, Mechatronics and Automation*, 2016, pp. 36–40.

[41] A. Ghaddar and A. Merei, “Energy-aware grid based coverage path planning for uavs,” in *Proceedings of the Thirteenth International Conference on Sensor Technologies and Applications SENSORCOMM, Nice, France*, IARIA. IARIA, October 27-31 2019, pp. 34–45.

[42] A. Ghaddar, A. Merei, and E. Natalizio, “Pps: Energy-aware grid-based coverage path planning for uavs using area partitioning in the presence of nfzs,” *Sensors*, vol. 20, no. 13, p. 3742, 2020.

[43] M. Torres, D. A. Pelta, J. L. Verdegay, and J. C. Torres, “Coverage path planning with unmanned aerial vehicles for 3d terrain reconstruction,” *Expert Systems with Applications*, vol. 55, pp. 441–451, 2016.

[44] T. M. Cabreira, C. Di Franco, P. R. Ferreira, and G. C. Buttazzo, “Energy-aware spiral coverage path planning for uav photogrammetric applications,” *IEEE Robotics and automation letters*, vol. 3, no. 4, pp. 3662–3668, 2018.

[45] F. Balampanis, I. Maza, and A. Ollero, “Area decomposition, partition and coverage with multiple remotely piloted aircraft systems operating in coastal regions,” in *2016*

International Conference on Unmanned Aircraft Systems (ICUAS), 07 2016, pp. 275–283.

[46] Balampanis, Fotios and Maza, Ivan and Ollero, Anibal, "Area partition for coastal regions with multiple uas," *Journal of Intelligent and Robotic Systems*, vol. 88, no. 2, pp. 751–766, Dec 2017. [Online]. Available: https://doi.org/10.1007/s10846-017-0559-9

[47] M. Li, A. Richards, and M. Sooriyabandara, "Reliability-aware multi-uav coverage path planning using integer linear programming," in *UKRAS20 Conference:" Robots into the real world" Proceedings. EPSRC UK-RAS Network*, 2020, pp. 15–17.

[48] D.-H. Cho, J.-S. Ha, S. Lee, S. Moon, and H.-L. Choi, "Informative path planning and mapping with multiple uavs in wind fields," in *Distributed Autonomous Robotic Systems*, T. Takahashi, J. Ota, and S. L. Smith, Eds. Springer, 2018, pp. 269–283.

[49] M. Popovic, G. Hitz, J. Nieto, I. Sa, R. Siegwart, and E. Galceran, "Online informative path planning for active classification using uavs," *arXiv preprint arXiv:1609.08446*, 2016.

[50] M. Popović, T. Vidal-Calleja, G. Hitz, I. Sa, R. Siegwart, and J. Nieto, "Multiresolution mapping and informative path planning for uav-based terrain monitoring," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 1382–1388.

[51] A. Blanchard and T. Sapsis, "Informative path planning for extreme anomaly detection in environment exploration and monitoring," *arXiv preprint arXiv:2005.10040*, 2020.

[52] T. M. Cabreira, P. R. Ferreira, C. Di Franco, and G. C. Buttazzo, "Grid-based coverage path planning with minimum energy over irregular-shaped areas with uavs," in *International conference on unmanned aircraft systems (ICUAS)*. IEEE, 2019, pp. 758–767.

[53] A. Ghaddar and A. Merei, "Eaoa: Energy-aware grid-based 3d-obstacle avoidance in coverage path planning for uavs," *Future Internet*, vol. 12, no. 2, p. 29, 2020.

[54] F. Yaghmaee and H. R. Koohi, "Dynamic obstacle avoidance by distributed algorithm based on reinforcement learning," *Int J Eng (IJE) Trans B: Appl*, vol. 28, no. 2, pp. 198–205, 2015.

[55] R. Zhao and D. Sidobre, "Dynamic obstacle avoidance using online trajectory time-scaling and local replanning," in *2015 12th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, vol. 2. IEEE, 2015, pp. 414–421.

[56] L. Zhao, "3d obstacle avoidance for unmanned autonomous system (uas)," Master's Thesis, University of Nevada, Las Vegas, 2015. [Online]. Available: https://digitalscholarship.unlv.edu/thesesdissertations/2507/

[57] C. Liu, H. Wang, and P. Yao, "Uav autonomous collision avoidance method based on three-dimensional dynamic collision region model and interfered fluid dynamical system'," in *INTERNATIONAL CONFERENCE ON CONTROL AND AUTOMATION*, 2016, pp. 263–269.

[58] S. Huang, R. S. H. Teo, and K. K. Tan, "Collision avoidance of multi unmanned aerial vehicles: A review," *Annual Reviews in Control*, vol. 48, pp. 147–164, 2019.

[59] X.-X. Shao, Y.-J. Gong, Z.-H. Zhan, and J. Zhang, "Bipartite cooperative coevolution for energy-aware coverage path planning of uavs," *IEEE Transactions on Artificial Intelligence*, vol. 3, no. 1, pp. 29–42, 2021.

[60] C. Zhan, H. Hu, X. Sui, Z. Liu, and D. Niyato, "Completion time and energy optimization in the uav-enabled mobile-edge computing system," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7808–7822, 2020.

[61] H. Wang, J. Wang, G. Ding, J. Chen, F. Gao, and Z. Han, "Completion time minimization with path planning for fixed-wing uav communications," *IEEE Transactions on Wireless Communications*, vol. 18, no. 7, pp. 3485–3499, 2019.

[62] M. R. Estrada and M. Arturo, "The uses of drones in case of massive epidemics contagious diseases relief humanitarian aid: Wuhan-covid-19 crisis," *Available at SSRN*, 2020.

[63] J. Valente, J. Del Cerro, A. Barrientos, and D. Sanz, "Aerial coverage optimization in precision agriculture management: A musical harmony inspired approach," *Computers and electronics in agriculture*, vol. 99, pp. 153–159, 2013.

[64] A. Barrientos, J. Colorado, J. d. Cerro, A. Martinez, C. Rossi, D. Sanz, and J. Valente,

"Aerial remote sensing in agriculture: A practical approach to area coverage and path planning for fleets of mini aerial robots," *Journal of Field Robotics*, vol. 28, no. 5, pp. 667–689, 2011.

[65] M. Krey and R. Seiler, "Usage and acceptance of drone technology in healthcare: exploring patients and physicians perspective," in *52nd Hawaii International Conference on System Sciences, Grand Wailea HI, USA, 8-11 January 2019*. HICSS, 2019, pp. 4135–4144.

[66] H. E. Comtet and K.-A. Johannessen, "The moderating role of pro-innovative leadership and gender as an enabler for future drone transports in healthcare systems," *International Journal of Environmental Research and Public Health*, vol. 18, no. 5, p. 2637, 2021.

[67] D. Cawthorne and A. Robbins-van Wynsberghe, "An ethical framework for the design, development, implementation, and assessment of drones used in public healthcare," *Science and Engineering Ethics*, vol. 26, no. 5, pp. 2867–2891, 2020.

[68] V. Jeyabalan, E. Nouvet, P. Meier, and L. Donelle, "Context-specific challenges, opportunities, and ethics of drones for healthcare delivery in the eyes of program managers and field staff: a multi-site qualitative study," *Drones*, vol. 4, no. 3, p. 44, 2020.

[69] M. A. H. Zailani, R. Z. A. R. Sabudin, R. A. Rahman, I. M. Saiboon, A. Ismail, and Z. A. Mahdy, "Drone for medical products transportation in maternal healthcare: A systematic review and framework for future research," *Medicine*, vol. 99, no. 36, 2020.

[70] T. Talaie, S. Niederhaus, E. Villalongas, and J. Scalea, "Innovating organ delivery to improve access to care: surgeon perspectives on the current system and future use of unmanned aircrafts," *BMJ Innovations*, vol. 7, no. 1, 2021.

[71] J. R. Scalea, S. Restaino, M. Scassero, G. Blankenship, S. T. Bartlett, and N. Wereley, "An initial investigation of unmanned aircraft systems (uas) and real-time organ status measurement for transporting human organs," *IEEE journal of translational engineering in health and medicine*, vol. 6, pp. 1–7, 2018.

[72] N. Balakrishnan, K. Devaraj, S. Rajan, and G. Seshadri, "Transportation of organs using uav," in *Proc. Int. Conf. Ind. Eng. Oper. Management*, vol. 3090, 2016.

[73] M. Al-Ayyad, A. Al-Ghraibah, and H. Alkhatib, “Transportation of donated hearts by drone in comparison to road-bound vehicles in mexico city,” *The Open Biomedical Engineering Journal*, vol. 13, no. 1, 2019.

[74] S. S. Fakhrulddin, S. K. Gharghan, A. Al-Naji, and J. Chahl, “An advanced first aid system based on an unmanned aerial vehicles and a wireless body area sensor network for elderly persons in outdoor environments,” *Sensors*, vol. 19, no. 13, p. 2955, 2019.

[75] M. Poljak and A. Šterbenc, “Use of drones in clinical microbiology and infectious diseases: current status, challenges and barriers,” *Clinical Microbiology and Infection*, vol. 26, no. 4, pp. 425–430, 2020.

[76] G. Ling and N. Draghic, “Aerial drones for blood delivery,” *Transfusion*, vol. 59, no. S2, pp. 1608–1611, 2019.

[77] B. McCall, “Sub-saharan africa leads the way in medical drones,” *The Lancet*, vol. 393, no. 10166, pp. 17–18, 2019.

[78] V. Homier, D. Brouard, M. Nolan, M.-A. Roy, P. Pelletier, M. McDonald, F. de Champlain, E. Khalil, F. Grou-Boileau, and R. Fleet, “Drone versus ground delivery of simulated blood products to an urban trauma center: the montreal medi-drone pilot study,” *The journal of trauma and acute care surgery*, vol. 90, no. 3, p. 515, 2021.

[79] S. Beck, T. T. Bui, A. Davies, P. Courtney, A. Brown, J. Geudens, and P. G. Royall, “An evaluation of the drone delivery of adrenaline auto-injectors for anaphylaxis: Pharmacists’ perceptions, acceptance, and concerns,” *Drones*, vol. 4, no. 4, p. 66, 2020.

[80] A. A. Nyaaba and M. Ayamga, “Intricacies of medical drones in healthcare delivery: Implications for africa,” *Technology in Society*, vol. 66, p. 101624, 2021.

[81] J. C. Rosser Jr, V. Vignesh, B. A. Terwilliger, and B. C. Parker, “Surgical and medical applications of drones: A comprehensive review,” *JSLS: Journal of the Society of Laparoendoscopic Surgeons*, vol. 22, no. 3, 2018.

[82] M. A. Khan, B. A. Alvi, A. Safi, and I. U. Khan, “Drones for good in smart cities: A review,” in *Proc. Int. Conf. Elect., Electron., Comput., Commun., Mech. Comput.(EECCMC)*, 2018,

pp. 1–6.

[83] K. B. Laksham, "Unmanned aerial vehicle (drones) in public health: A swot analysis," *Journal of family medicine and primary care*, vol. 8, no. 2, p. 342, 2019.

[84] S. Schierbeck, J. Hollenberg, A. Nord, L. Svensson, P. Nordberg, M. Ringh, S. Forsberg, P. Lundgren, C. Axelsson, and A. Claesson, "Automated external defibrillators delivered by drones to patients with suspected out-of-hospital cardiac arrest," *European Heart Journal*, vol. 42, no. Supplement_1, pp. ehab724–0656, 2021.

[85] C. Wankmüller, C. Truden, C. Korzen, P. Hungerländer, E. Kolesnik, and G. Reiner, "Optimal allocation of defibrillator drones in mountainous regions," *OR Spectrum*, vol. 42, no. 3, pp. 785–814, 2020.

[86] J. J. Boutilier, S. C. Brooks, A. Janmohamed, A. Byers, J. E. Buick, C. Zhan, A. P. Schoellig, S. Cheskes, L. J. Morrison, and T. C. Chan, "Optimizing a drone network to deliver automated external defibrillators," *Circulation*, vol. 135, no. 25, pp. 2454–2465, 2017.

[87] J. J. Boutilier and T. C. Chan, "Response time optimization for drone-delivered automated external defibrillators," *arXiv preprint arXiv:1908.00149*, 2019.

[88] S. Cheskes, S. L. McLeod, M. Nolan, P. Snobelen, C. Vaillancourt, S. C. Brooks, K. N. Dainty, T. C. Chan, and I. R. Drennan, "Improving access to automated external defibrillators in rural and remote settings: a drone delivery feasibility study," *Journal of the American Heart Association*, vol. 9, no. 14, p. e016687, 2020.

[89] A. Pulver, R. Wei, and C. Mann, "Locating aed enabled medical drones to enhance cardiac arrest response times," *Prehospital Emergency Care*, vol. 20, no. 3, pp. 378–389, 2016.

[90] A. Claesson, D. Fredman, L. Svensson, M. Ringh, J. Hollenberg, P. Nordberg, M. Rosenqvist, T. Djarv, S. Österberg, J. Lennartsson *et al.*, "Unmanned aerial vehicles (drones) in out-of-hospital-cardiac-arrest," *Scandinavian journal of trauma, resuscitation and emergency medicine*, vol. 24, no. 1, pp. 1–9, 2016.

[91] S. Sarker, L. Jamal, S. F. Ahmed, and N. Irtisam, "Robotics and artificial intelligence in

healthcare during covid-19 pandemic: A systematic review," *Robotics and autonomous systems*, vol. 146, p. 103902, 2021.

[92] E. Mbunge, I. Chitungo, and T. Dzinamarira, "Unbundling the significance of cognitive robots and drones deployed to tackle covid-19 pandemic: A rapid review to unpack emerging opportunities to improve healthcare in sub-saharan africa," *Cognitive Robotics*, vol. 1, pp. 205–213, 2021.

[93] J. Euchi, "Do drones have a realistic place in a pandemic fight for delivering medical supplies in healthcare systems problems?" pp. 182–190, 2021.

[94] Z. H. Khan, A. Siddique, and C. W. Lee, "Robotics utilization for healthcare digitization in global covid-19 management," *International journal of environmental research and public health*, vol. 17, no. 11, p. 3819, 2020.

[95] A. Oigbochie, E. Odigie, and B. Adejumo, "Importance of drones in healthcare delivery amid a pandemic: Current and generation next application," *Open Journal of Medical Research (ISSN: 2734-2093)*, vol. 2, no. 1, pp. 01–13, 2021.

[96] A. Alsarhan, I. T. Almalkawi, and Y. Kilani, "New covid-19 tracing approach using machine learning and drones enabled wireless network." *International Journal of Interactive Mobile Technologies*, vol. 15, no. 22, 2021.

[97] A. Kumar, K. Sharma, H. Singh, S. G. Naugriya, S. S. Gill, and R. Buyya, "A drone-based networked system and methods for combating coronavirus disease (covid-19) pandemic," *Future Generation Computer Systems*, vol. 115, pp. 1–19, 2021.

[98] Y. Karaca, M. Cicek, O. Tatli, A. Sahin, S. Pasli, M. F. Beser, and S. Turedi, "The potential use of unmanned aircraft systems (drones) in mountain search and rescue operations," *The American journal of emergency medicine*, vol. 36, no. 4, pp. 583–588, 2018.

[99] P. Doherty and P. Rudol, "A uav search and rescue scenario with human body detection and geolocalization," in *Australasian Joint Conference on Artificial Intelligence*. Springer, 2007, pp. 1–13.

[100] D. Câmara, "Cavalry to the rescue: Drones fleet to help rescuers operations over disasters

scenarios," in *2014 IEEE Conference on Antenna Measurements & Applications (CAMA)*. IEEE, 2014, pp. 1–4.

[101] V. Mersheeva and G. Friedrich, "Routing for continuous monitoring by multiple micro avs in disaster scenarios," in *Proceedings of the 20th European Conference on Artificial Intelligence (ECAI 2012)*. IOS Press, 2012, pp. 588–593.

[102] J. Q. Cui, S. K. Phang, K. Z. Ang, F. Wang, X. Dong, Y. Ke, S. Lai, K. Li, X. Li, F. Lin *et al.*, "Drones for cooperative search and rescue in post-disaster situation," in *2015 IEEE 7th international conference on cybernetics and intelligent systems (CIS) and IEEE conference on robotics, automation and mechatronics (RAM)*. IEEE, 2015, pp. 167–174.

[103] B. Wimberley, "The use of drones in healthcare," Online, Oct 2021, available: https://healthcaretransformers.com/healthcare-business/drones-healthcare/.

[104] R. F. Graboyes and B. Skorup, "Medical drones in the united states and a survey of technical and policy challenges," *Mercatus Center Policy Brief*, 2020.

[105] B. Shruthi, K. Manasa, and R. Lakshmi, "Survey on challenges and future scope of iot in healthcare and agriculture," *International Journal of Computer Science and Mobile Computing*, vol. 8, no. 1, pp. 21–26, 2019.

[106] L. Ersson and E. Olsson, "Drones to the rescue: A literary study of unmanned aerial systems within healthcare," Online, 2020, available: URL_PLACEHOLDER.

[107] D. A. Allan-Matheson, "Unmanned aerial vehicles in healthcare: Now and for the future – investing into emerging technology for rapid vaccination delivery in healthcare, globally and especially in rural africa," Master's paper, University of North Carolina at Chapel Hill, May 2018. [Online]. Available: https://cdr.lib.unc.edu/concern/masters_papers/pr76f6784

[108] D. Sachan, "The age of drones: what might it mean for health?" *Lancet (London, England)*, vol. 387, no. 10030, pp. 1803–1804, 2016.

[109] J. Ingham, "Drones: Could they be used to deliver blood?" Online, Jan 2020, available: https://www.bbc.com/news/av/uk-england-hampshire-50960710.

[110] N. Ayyappaa, A. Y. Raj, A. Adithya, R. Murali, and A. Vinodh, “Autonomous drone for efficacious blood conveyance,” in *2019 4th International Conference on Robotics and Automation Engineering (ICRAE)*. IEEE, 2019, pp. 99–103.

[111] T. Wen, Z. Zhang, and K. K. Wong, “Multi-objective algorithm for blood supply via unmanned aerial vehicles to the wounded in an emergency situation,” *PloS one*, vol. 11, no. 5, p. e0155176, 2016.

[112] T. K. Amukele, J. Hernandez, C. L. Snozek, R. G. Wyatt, M. Douglas, R. Amini, and J. Street, “Drone transport of chemistry and hematology samples over long distances,” *American journal of clinical pathology*, vol. 148, no. 5, pp. 427–435, 2017.

[113] T. Amukele, P. M. Ness, A. A. Tobian, J. Boyd, and J. Street, “Drone transportation of blood products,” *Transfusion*, vol. 57, no. 3, pp. 582–588, 2017.

[114] W. H. Organization, “Coronavirus (covid-19) dashboard,” Online, Jan 2022, available: https://covid19.who.int/.

[115] P. K. Dutta and S. Mitra, “Application of agricultural drones and iot to understand food supply chain during post covid-19,” *Agricultural Informatics: Automation Using the IoT and Machine Learning*, pp. 67–87, 2021.

[116] E. Lamptey and D. Serwaa, “The use of zipline drones technology for covid-19 samples transportation in ghana,” *HighTech and Innovation Journal*, vol. 1, no. 2, pp. 67–71, 2020.

[117] M. A. Pinheiro, M. Liu, Y. Wan, and A. Dogan, “On the analysis of on-board sensing and off-board sensing through wireless communication for uav path planning in wind fields,” in *AIAA Scitech 2019 Forum*, 2019, p. 2131.

[118] S.-H. Kim, G. E. G. Padilla, K.-J. Kim, and K.-H. Yu, “Flight path planning for a solar powered uav in wind fields using direct collocation,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 2, pp. 1094–1105, 2019.

[119] H. Chen, P. Lu, and C. Xiao, “Dynamic obstacle avoidance for uavs using a fast trajectory planning approach,” in *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2019, pp. 1459–1464.

[120] N. E.-D. Safwat, I. M. Hafez, and F. Newagy, “3d placement of a new tethered uav to uav relay system for coverage maximization,” *Electronics*, vol. 11, no. 3, p. 385, 2022.

[121] Y. Wang, X. Zhu, and L. Xu, “Flight path optimization for uavs to provide location service to ground targets,” in *2020 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2020, pp. 1–6.

[122] X. Wu, W. Bai, Y. Xie, X. Sun, C. Deng, and H. Cui, “A hybrid algorithm of particle swarm optimization, metropolis criterion and rts smoother for path planning of uavs,” *Applied Soft Computing*, vol. 73, pp. 735–747, 2018.

[123] H. Yang, J. Zhang, S. Song, and K. B. Lataief, “Connectivity-aware uav path planning with aerial coverage maps,” in *2019 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2019, pp. 1–6.

[124] J. Scott and C. Scott, “Drone delivery models for healthcare,” in *Proceedings of the 50th Hawaii international conference on system sciences*, 2017.

[125] A. Abid, S. Cheikhrouhou, S. Kallel, and M. Jmaiel, “Temporal constraints in smart contract-based process execution: A case study of organ transfer by healthcare delivery drone,” in *Proceedings of the Tunisian-Algerian Joint Conference on Applied Computing (TACC 2021)*, A. Hadj Kacem, S. Kallel, F. Belala, M. Belguidoum, M. Jmaiel, and I. Bouassida Rodriguez, Eds., vol. 3067. CEUR-WS.org, 2021, pp. 1–13. [Online]. Available: https://ceur-ws.org/Vol-3067/paper1.pdf

[126] J. Li, W. Goh, and N. Jhanjhi, “A design of iot-based medicine case for the multi-user medication management using drone in elderly centre,” *Journal of Engineering Science and Technology*, vol. 16, no. 2, pp. 1145–1166, 2021.

[127] G. D. Kumar and B. Jeeva, “Drone ambulance for outdoor sports,” *Asian J. Appl. Sci. and Technol*, vol. 1, pp. 44–49, 2017.

[128] E. V. Vazquez-Carmona, J. I. Vasquez-Gomez, J. C. H. Lozada, and M. Antonio-Cruz, “Coverage path planning for spraying drones,” *arXiv preprint arXiv:2105.08743*, 2021.

[129] F. Saeed, A. Mehmood, M. F. Majeed, C. Maple, K. Saeed, M. K. Khattak, H. Wang, and

G. Epiphaniou, “Smart delivery and retrieval of swab collection kit for covid-19 test using autonomous unmanned aerial vehicles,” *Physical Communication*, vol. 48, p. 101373, 2021.

[130] S. I. Khan, Z. Qadir, H. S. Munawar, S. R. Nayak, A. K. Budati, K. D. Verma, and D. Prakash, “Uavs path planning architecture for effective medical emergency response in future networks,” *Physical Communication*, vol. 47, p. 101337, 2021.

[131] A. Bitar, A. Jamal, H. Sultan, N. Alkandari, and M. El-Abd, “Medical drones system for amusement parks,” in *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*. IEEE, 2017, pp. 19–20.

[132] A. Singh, P. Kumar, K. Pachauri, and K. Singh, “Drone ambulance,” in *2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*. IEEE, 2020, pp. 705–708.

[133] R. K. Rangel, “Development of low cost medical drone, using cots equipment,” in *2021 IEEE Aerospace Conference (50100)*. IEEE, 2021, pp. 1–12.

[134] A. J. A. Dhivya and J. Premkumar, “Quadcopter based technology for an emergency healthcare,” in *2017 Third International Conference on Biosignals, Images and Instrumentation (ICBSII)*. IEEE, 2017, pp. 1–3.

[135] P. Sanjana and M. Prathilothamai, “Drone design for first aid kit delivery in emergency situation,” in *2020 6th international conference on advanced computing and communication systems (ICACCS)*. IEEE, 2020, pp. 215–220.

[136] P. Kulp and N. Mei, “A framework for sensing radio frequency spectrum attacks on medical delivery drones,” in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2020, pp. 408–413.

[137] L. Faramondi, G. Oliva, L. Ardito, A. Crescenzi, M. Caricato, M. Tesei, A. O. Muda, and R. Setola, “Use of drone to improve healthcare efficiency and sustainability,” in *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*. IEEE, 2020, pp. 1783–1788.

[138] N. Nithyavathy, S. Pavithra, M. Naveen, B. Logesh, and T. James, “Design and development

of drone for healthcare," *Int. J. Sci. Technol. Res*, vol. 9, no. 1, pp. 2676–2680, 2020.

[139] M. S. Y. Hii, P. Courtney, and P. G. Royall, "An evaluation of the delivery of medicines using drones," *Drones*, vol. 3, no. 3, p. 52, 2019.

[140] T. K. Amukele, J. Street, K. Carroll, H. Miller, and S. X. Zhang, "Drone transport of microbes in blood and sputum laboratory specimens," *Journal of clinical microbiology*, vol. 54, no. 10, pp. 2622–2625, 2016.

[141] M. E. Nenni, V. Di Pasquale, S. Miranda, and S. Riemma, "Development of a drone-supported emergency medical service," *International Journal of Technology (IJTech)*, vol. 11, no. 4, 2020.

[142] S. Ullah, K.-I. Kim, K. H. Kim, M. Imran, P. Khan, E. Tovar, and F. Ali, "Uav-enabled healthcare architecture: Issues and challenges," *Future Generation Computer Systems*, vol. 97, pp. 425–432, 2019.

[143] J. N. Yasin, S. A. Mohamed, M.-H. Haghbayan, J. Heikkonen, H. Tenhunen, and J. Plosila, "Unmanned aerial vehicles (uavs): Collision avoidance systems and approaches," *IEEE access*, vol. 8, pp. 105 139–105 155, 2020.

[144] H. Pham, S. A. Smolka, S. D. Stoller, D. Phan, and J. Yang, "A survey on unmanned aerial vehicle collision avoidance systems," *arXiv preprint arXiv:1508.07723*, 2015.

[145] J. Tang, S. Lao, and Y. Wan, "Systematic review of collision-avoidance approaches for unmanned aerial vehicles," *IEEE Systems Journal*, 2021.

[146] K. Dushime, L. Nkenyereye, S. K. Yoo, and J. Song, "A review on collision avoidance systems for unmanned aerial vehicles," in *International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2021, pp. 1150–1155.

[147] H. T. Do, H. T. Hua, M. T. Nguyen, C. V. Nguyen, H. T. Nguyen, H. T. Nguyen, and N. T. Nguyen, "Formation control algorithms for multiple-uavs: a comprehensive survey," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, vol. 8, no. 27, pp. e3–e3, 2021.

[148] A. Sharma, S. Shoval, A. Sharma, and J. K. Pandey, "Path planning for multiple targets interception by the swarm of uavs based on swarm intelligence algorithms: A review," *IETE Technical Review*, vol. 39, no. 3, pp. 675–697, 2022.

[149] Z. Kewang and D. Tenghuan, "Research on obstacle avoidance control method of multi-uav based on model predictive control," in *International Conference on Electronics, Circuits and Information Engineering (ECIE)*. IEEE, 2021, pp. 357–362.

[150] Y. Wu, J. Gou, X. Hu, and Y. Huang, "A new consensus theory-based method for formation control and obstacle avoidance of uavs," *Aerospace Science and Technology*, vol. 107, p. 106332, 2020.

[151] S. Ann, Y. Kim, and J. Ahn, "Area allocation algorithm for multiple uavs area coverage based on clustering and graph method," *IFAC-PapersOnLine*, vol. 48, no. 9, pp. 204–209, 2015.

[152] J.-A. Delamer, Y. Watanabe, and C. P. Chanel, "Safe path planning for uav urban operation under gnss signal occlusion risk," *Robotics and Autonomous Systems*, vol. 142, p. 103800, 2021.

[153] Ucl, "Depthmapx: Visual and spatial network analysis software," *The Bartlett School of Architecture*, Dec 2020, available at https://www.ucl.ac.uk/bartlett/architecture/research/space-syntax/depthmapx.

[154] A. Singla, S. Padakandla, and S. Bhatnagar, "Memory-based deep reinforcement learning for obstacle avoidance in uav with limited environment knowledge," *IEEE transactions on intelligent transportation systems*, vol. 22, no. 1, pp. 107–118, 2019.

[155] S. Vanneste, B. Bellekens, and M. Weyn, "3dvfh+: Real-time three-dimensional obstacle avoidance using an octomap," in *MORSE Model-Driven Robot Software Engineering: proceedings of the 1st International Workshop on Model-Driven Robot Software Engineering co-located with International Conference on Software Technologies: Applications and Foundations (STAF 2014), York, UK, July 21, 2014/Assmann, Uwe [edit.]*, no. 1319, 2014, pp. 91–102.

[156] M. Labbé and F. Michaud, "Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation," *Journal*

of field robotics, vol. 36, no. 2, pp. 416–446, 2019.

[157] W. Wu, B. Chen, and D. Liu, "Simulation of uav-based indoor mapping for multi-story building using a rgb-d slam solution," in *2024 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. IEEE, 2024, pp. 1–6.

[158] A. C. Plascencia, V. Beran, and K. Sedlmajer, "Drone sensory data processing for advanced drone control for augmented reality," Brno University of Technology, Brno, Czech Republic, Technical Report VI20172020068, 2019. [Online]. Available: https://www.fit.vut.cz/research/publication/c168681/.en

[159] A. Elfes, "Occupancy grids: A stochastic spatial representation for active robot perception," *arXiv preprint arXiv:1304.1098*, 2013.

[160] J. J. Tai, S. K. Phang, and F. Y. M. Wong, "Coaa*—an optimized obstacle avoidance and navigational algorithm for uavs operating in partially observable 2d environments," *Unmanned Systems*, vol. 10, no. 02, pp. 159–174, 2022.

[161] P. Peng, W. Dong, G. Chen, and X. Zhu, "Obstacle avoidance of resilient uav swarm formation with active sensing system in the dense environment," *arXiv preprint arXiv:2202.13381*, 2022.

[162] Baeldung, "Disparity map in stereo vision," *Baeldung on Computer Science*, May 2022, available at https://www.baeldung.com/cs/disparity-map-stereo-vision.

[163] Asier03, "Category:textures," *Wikimedia Commons*, Nov 2019, available at https://commons.wikimedia.org/wiki/Category:Textures.

[164] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, 2013, software available at https://octomap.github.io. [Online]. Available: https://octomap.github.io

[165] B. Ruf, S. Monka, M. Kollmann, and M. Grinberg, "Real-time on-board obstacle avoidance for uavs based on embedded stereo vision," *arXiv preprint arXiv:1807.06271*, 2018.

[166] D. Han, Q. Yang, and R. Wang, "Three-dimensional obstacle avoidance for uav based on reinforcement learning and realsense," *The Journal of Engineering*, vol. 2020, no. 13, pp. 540–544, 2020.

[167] Z. Yaoming, S. Yu, X. Anhuan, and K. Lingyu, "A newly bio-inspired path planning algorithm for autonomous obstacle avoidance of uav," *Chinese Journal of Aeronautics*, vol. 34, no. 9, pp. 199–209, 2021.

[168] M. Iacono and A. Sgorbissa, "Path following and obstacle avoidance for an autonomous uav using a depth camera," *Robotics and Autonomous Systems*, vol. 106, pp. 38–46, 2018.

[169] Z. Chen, X. Luo, and B. Dai, "Design of obstacle avoidance system for micro-uav based on binocular vision," in *International Conference on Industrial Informatics-Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII)*. IEEE, 2017, pp. 67–70.

[170] T. Stefansson, "3d obstacle avoidance for drones using a realistic sensor setup," Online, 2018, available: https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1242725&dswid=-5894.

[171] E. Ferrera, A. Alcántara, J. Capitán, A. R. Castaño, P. J. Marrón, and A. Ollero, "Decentralized 3d collision avoidance for multiple uavs in outdoor environments," *Sensors*, vol. 18, no. 12, p. 4101, 2018.

[172] J. Roghair, A. Niaraki, K. Ko, and A. Jannesari, "A vision based deep reinforcement learning algorithm for uav obstacle avoidance," in *Proceedings of SAI Intelligent Systems Conference*. Springer, 2021, pp. 115–128.

[173] Z. Xue and T. Gonsalves, "Vision based drone obstacle avoidance by deep reinforcement learning," *AI*, vol. 2, no. 3, pp. 366–380, 2021.

[174] Y. Yu, W. Tingting, C. Long, and Z. Weiwei, "Stereo vision based obstacle avoidance strategy for quadcopter uav," in *Chinese Control And Decision Conference (CCDC)*. IEEE, 2018, pp. 490–494.

[175] A. Al-Kaff, F. García, D. Martín, A. De La Escalera, and J. M. Armingol, "Obstacle

detection and avoidance system based on monocular camera and size expansion algorithm for uavs," *Sensors*, vol. 17, no. 5, p. 1061, 2017.

[176] H. Yu, F. Zhang, P. Huang, C. Wang, and L. Yuanhao, "Autonomous obstacle avoidance for uav based on fusion of radar and monocular camera," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 5954–5961.

[177] T. Mori and S. Scherer, "First results in detecting and avoiding frontal obstacles from a monocular camera for micro unmanned aerial vehicles," in *IEEE International Conference on Robotics and Automation*. IEEE, 2013, pp. 1750–1757.

[178] B. Park and H. Oh, "Vision-based obstacle avoidance for uavs via imitation learning with sequential neural networks," *International Journal of Aeronautical and Space Sciences*, vol. 21, no. 3, pp. 768–779, 2020.

[179] H. Lee, H. W. Ho, and Y. Zhou, "Deep learning-based monocular obstacle avoidance for unmanned aerial vehicle navigation in tree plantations," *Journal of Intelligent & Robotic Systems*, vol. 101, no. 1, pp. 1–18, 2021.

[180] M. C. Santos, L. V. Santana, A. S. Brandao, and M. Sarcinelli-Filho, "Uav obstacle avoidance using rgb-d system," in *2015 international conference on unmanned aircraft systems (ICUAS)*. IEEE, 2015, pp. 312–319.

[181] D. Wang, W. Li, X. Liu, N. Li, and C. Zhang, "Uav environmental perception and autonomous obstacle avoidance: A deep learning and depth camera combined solution," *Computers and Electronics in Agriculture*, vol. 175, p. 105523, 2020.

[182] P. Kuncha, A. S. C. S. Sastry, and M. Chaitanya, "Dynamic route planning and obstacle avoidance model for unmanned aerial vehicles," *International Journal of Pure and Applied Mathematics*, vol. 116, no. 24, pp. 315–329, November 2017. [Online]. Available: https://www.researchgate.net/publication/321018379_Dynamic_Route_Planning_and_Obstacle_Avoidance_Model_for_Unmanned_Aerial_Vehicles

[183] P. B. Parappat, A. Kumar, R. Mittal, and S. A. Khan, "Obstacle avoidance by unmanned aerial vehicles using image recognition techniques," in *Proceedings of the International Conference on Circuits, Systems, Communications and Computers*, vol. 1, 2014, pp. 378–381.

[184] S.-Y. Shin, Y.-W. Kang, and Y.-G. Kim, "Obstacle avoidance drone by deep reinforcement learning and its racing with human pilot," *Applied sciences*, vol. 9, no. 24, p. 5571, 2019.

[185] D.-H. Kim, Y.-G. Go, and S.-M. Choi, "First-person-view drone flying in mixed reality," in *SIGGRAPH Asia 2018 Posters*, 2018, pp. 1–2.

[186] N. Smolyanskiy and M. Gonzalez-Franco, "Stereoscopic first person view system for drone navigation," *Frontiers in Robotics and AI*, p. 11, 2017.

[187] B. B. Tierney and C. T. Rodenbeck, "3d-sensing mimo radar for uav formation flight and obstacle avoidance," in *IEEE Radio and Wireless Symposium (RWS)*. IEEE, 2019, pp. 1–3.

[188] C. Potena, D. Nardi, and A. Pretto, "Joint vision-based navigation, control and obstacle avoidance for uavs in dynamic environments," in *European Conference on Mobile Robots (ECMR)*. IEEE, 2019, pp. 1–7.

[189] Y. Xu, M. Zhu, Y. Xu, and M. Liu, "Design and implementation of uav obstacle avoidance system," in *2nd International Conference on Safety Produce Informatization (IICSPI)*. IEEE, 2019, pp. 275–278.

[190] N. Gageik, P. Benz, and S. Montenegro, "Obstacle detection and collision avoidance for a uav with complementary low-cost sensors," *IEEE Access*, vol. 3, pp. 599–609, 2015.

[191] K. Gao and K. Xu, "Research on autonomous obstacle avoidance flight path planning of rotating wing uav," in *International Conference on Computer Technology, Electronics and Communication (ICCTEC)*. IEEE, 2017, pp. 1200–1203.

[192] T. Kenny, "Chapter 14 - optical and radiation sensors," in *Sensor Technology Handbook*, J. S. Wilson, Ed. Burlington: Newnes, 2005, pp. 307–320. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780750677295500549

[193] M. J. McGrath and C. N. Scanaill, *Key Sensor Technology Components: Hardware and Software Overview*. Berkeley, CA: Apress, 2013, pp. 51–77. [Online]. Available: https://doi.org/10.1007/978-1-4302-6014-1_3

[194] J. S. Wilson, “Chapter 1 - sensor fundamentals,” in *Sensor Technology Handbook*, J. S. Wilson, Ed. Burlington: Newnes, 2005, pp. 1–20. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780750677295500410

[195] Jon S. Wilson, “Chapter 3 - measurement issues and criteria,” in *Sensor Technology Handbook*, J. S. Wilson, Ed. Burlington: Newnes, 2005, pp. 29–30. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780750677295500434

[196] W. Kester, “Chapter 4 - sensor signal conditioning,” in *Sensor Technology Handbook*, J. S. Wilson, Ed. Burlington: Newnes, 2005, pp. 31–136. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780750677295500446

[197] C. Aszkler, “Chapter 5 - acceleration, shock and vibration sensors,” in *Sensor Technology Handbook*, J. S. Wilson, Ed. Burlington: Newnes, 2005, pp. 137–159. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780750677295500458

[198] J. S. Wilson, “Chapter 15 - position and motion sensors,” in *Sensor Technology Handbook*, J. S. Wilson, Ed. Burlington: Newnes, 2005, pp. 321–409. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780750677295500550

[199] N. Bashir, S. Boudjit, G. Dauphin, and S. Zeadally, “An obstacle avoidance approach for uav path planning,” *Simulation modelling practice and theory*, vol. 129, p. 102815, 2023.

[200] Y. Lin, F. Gao, T. Qin, W. Gao, T. Liu, W. Wu, Z. Yang, and S. Shen, “Autonomous aerial navigation using monocular visual-inertial fusion,” *Journal of Field Robotics*, vol. 35, no. 1, pp. 23–51, 2018.

[201] Z. Xu, X. Zhan, B. Chen, Y. Xiu, C. Yang, and K. Shimada, “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. 10 645–10 651.

[202] H. Du, Z. Wang, and X. Zhang, “Ef-ttoa: Development of a uav path planner and obstacle avoidance control framework for static and moving obstacles,” *Drones*, vol. 7, no. 6, p. 359, 2023.

[203] Q. Geng, H. Shuai, and Q. Hu, "Obstacle avoidance approaches for quadrotor uav based on backstepping technique," in *25th Chinese Control and Decision Conference (CCDC)*. IEEE, 2013, pp. 3613–3617.

[204] M. Choi, A. Rubenecia, T. Shon, and H. H. Choi, "Velocity obstacle based 3d collision avoidance scheme for low-cost micro uavs," *Sustainability*, vol. 9, no. 7, p. 1174, 2017.

[205] Z. Zhang, X. Liu, and B. Feng, "Research on obstacle avoidance path planning of uav in complex environments based on improved bézier curve," *Scientific Reports*, vol. 13, no. 1, p. 16453, 2023.

[206] A. Chakravarthy and D. Ghose, "Obstacle avoidance in a dynamic environment: A collision cone approach," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 28, no. 5, pp. 562–574, 1998.

[207] Chakravarthy, Animesh and D. Ghose, "Generalization of the collision cone approach for motion safety in 3-d environments," *Autonomous Robots*, vol. 32, pp. 243–266, 2012.

[208] L. Agnel Tony, D. Ghose, and A. Chakravarthy, "Unmanned aerial vehicle mid-air collision detection and resolution using avoidance maps," *Journal of Aerospace Information Systems*, vol. 18, no. 8, pp. 506–529, 2021.

[209] L. A. Tony, D. Ghose, and A. Chakravarthy, "Avoidance maps: A new concept in uav collision avoidance," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2017, pp. 1483–1492. [Online]. Available: https://ieeexplore.ieee.org/document/7991372

[210] Z. Ma, Z. Wang, A. Ma, Y. Liu, and Y. Niu, "A low-altitude obstacle avoidance method for uavs based on polyhedral flight corridor," *Drones*, vol. 7, no. 9, p. 588, 2023.

[211] M. A. Trujillo, H. M. Becerra, D. Gómez-Gutiérrez, J. Ruiz-León, and A. Ramírez-Treviño, "Hierarchical task-based formation control and collision avoidance of uavs in finite time," *European Journal of Control*, vol. 60, pp. 48–64, 2021.

[212] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer vision and image understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[213] J. Hu, Y. Niu, and Z. Wang, "Obstacle avoidance methods for rotor uavs using realsense camera," in *2017 Chinese Automation Congress (CAC)*. IEEE, 2017, pp. 7151–7155.

[214] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, vol. 3. IEEE, 2004, pp. 2149–2154.

[215] W. Jiang, T. Cai, G. Xu, and Y. Wang, "Autonomous obstacle avoidance and target tracking of uav: Transformer for observation sequence in reinforcement learning," *Knowledge-Based Systems*, vol. 290, p. 111604, 2024.

[216] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[217] N. Aswini and S. Uma, "Custom based obstacle detection using yolo v3 for low flying drones," in *2021 International Conference on Circuits, Controls and Communications (CCUBE)*. IEEE, 2021, pp. 1–6.

[218] C. Jiang, H. Ren, X. Ye, J. Zhu, H. Zeng, Y. Nan, M. Sun, X. Ren, and H. Huo, "Object detection from uav thermal infrared images and videos using yolo models," *International Journal of Applied Earth Observation and Geoinformation*, vol. 112, p. 102912, 2022.

[219] X. Han, J. Wang, J. Xue, and Q. Zhang, "Intelligent decision-making for 3-dimensional dynamic obstacle avoidance of uav based on deep reinforcement learning," in *11th International conference on wireless communications and signal processing (WCSP)*. IEEE, 2019, pp. 1–6.

[220] A. Sonny, S. R. Yeduri, and L. R. Cenkeramaddi, "Q-learning-based unmanned aerial vehicle path planning with dynamic obstacle avoidance," *Applied Soft Computing*, vol. 147, p. 110773, 2023.

[221] C. Wang, Z. Wei, W. Jiang, H. Jiang, and Z. Feng, "Cooperative sensing enhanced uav path-following and obstacle avoidance with variable formation," *IEEE Transactions on Vehicular Technology*, 2024.

[222] X. Dai, Y. Mao, T. Huang, N. Qin, D. Huang, and Y. Li, "Automatic obstacle avoidance of quadrotor uav via cnn-based learning," *Neurocomputing*, vol. 402, pp. 346–358, 2020.

[223] M. Odelga, P. Stegagno, and H. H. Bülthoff, "Obstacle detection, tracking and avoidance for a teleoperated uav," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 2984–2990.

[224] Y. Feng, Y. Wu, H. Cao, and J. Sun, "Uav formation and obstacle avoidance based on improved apf," in *210th International Conference on Modelling, Identification and Control (ICMIC)*. IEEE, 2018, pp. 1–6.

[225] A. Ma'Arif, W. Rahmaniar, M. A. M. Vera, A. A. Nuryono, R. Majdoubi, and A. Çakan, "Artificial potential field algorithm for obstacle avoidance in uav quadrotor for dynamic environment," in *IEEE International Conference on Communication, Networks and Satellite (COMNETSAT)*. IEEE, 2021, pp. 184–189.

[226] Y. Liu, C. Chen, Y. Wang, T. Zhang, and Y. Gong, "A fast formation obstacle avoidance algorithm for clustered uavs based on artificial potential field," *Aerospace Science and Technology*, p. 108974, 2024.

[227] L. Keyu, L. Yonggen, and Z. Yanchi, "Dynamic obstacle avoidance path planning of uav based on improved apf," in *5th International Conference on Communication, Image and Signal Processing (CCISP)*. IEEE, 2020, pp. 159–163.

[228] Y. Zhao, L.-Y. Hao, and Z.-J. Wu, "Obstacle avoidance control of unmanned aerial vehicle with motor loss-of-effectiveness fault based on improved artificial potential field," *Sustainability*, vol. 15, no. 3, p. 2368, 2023.

[229] J. Li, J. J. Liu, P. Cheng, C. Liu, Y. Zhang, and B. Chen, "Event-based obstacle avoidance control for time-varying uav formation under cyber-attacks," *Journal of the Franklin Institute*, vol. 361, no. 13, p. 107019, 2024.

[230] X. Fu, C. Zhi, and D. Wu, "Obstacle avoidance and collision avoidance of uav swarm based on improved vfh algorithm and information sharing strategy," *Computers & Industrial Engineering*, vol. 186, p. 109761, 2023.

[231] X. Wang, M. Cheng, S. Zhang, and H. Gong, "Multi-uav cooperative obstacle avoidance of 3d vector field histogram plus and dynamic window approach," *Drones*, vol. 7, no. 8, p. 504, 2023.

[232] T. Wakabayashi, Y. Suzuki, and S. Suzuki, "Dynamic obstacle avoidance for multi-rotor uav using chance-constraints based on obstacle velocity," *Robotics and Autonomous Systems*, vol. 160, p. 104320, 2023.

[233] Y. Gong, K. Chen, T. Niu, and Y. Liu, "Grid-based coverage path planning with nfz avoidance for uav using parallel self-adaptive ant colony optimization algorithm in cloud iot," *Journal of Cloud Computing*, vol. 11, no. 1, pp. 1–28, 2022.

[234] P. Zhu, L. Qin, J. Wang, Y. Li, X. Li, and W. Xie, "Optimized trajectory and passive beamforming for star-ris-assisted uav-empowered o2i wpcn," *IEEE Wireless Communications Letters*, 2023.

[235] L. Sun, J. Wang, J. Wang, L. Lin, and M. Gen, "Efficient joint deployment of multi-uavs for target tracking in traffic big data," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[236] Y. Wu, T. Liang, J. Gou, C. Tao, and H. Wang, "Heterogeneous mission planning for multiple uav formations via metaheuristic algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 4, pp. 3924–3940, 2023.

[237] W. Xu, C. Wang, H. Xie, W. Liang, H. Dai, Z. Xu, Z. Wang, B. Guo, and S. K. Das, "Reward maximization for disaster zone monitoring with heterogeneous uavs," *IEEE/ACM Transactions on Networking*, 2023.

[238] Y. Gao, S. Wang, M. Liu, and Y. Hu, "Multi-agent reinforcement learning for uavs 3d trajectory designing and mobile ground users scheduling with no-fly zones," in *2023 IEEE/CIC International Conference on Communications in China (ICCC)*. IEEE, 2023, pp. 1–6.

[239] F. Trotti, A. Farinelli, and R. Muradore, "A markov decision process approach for decentralized uav formation path planning," in *2024 European Control Conference (ECC)*. IEEE, 2024, pp. 436–441.

[240] W. Lee and K. Lee, "Robust trajectory and resource allocation for uav communications in uncertain environments with no-fly zone: A deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[241] A. Andreou, C. X. Mavromoustakis, J. M. Batalla, E. K. Markakis, G. Mastorakis, and S. Mumtaz, "Uav trajectory optimisation in smart cities using modified a* algorithm combined with haversine and vincenty formulas," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 8, pp. 9757–9769, 2023.

[242] V. Vannini, G. de Moura Souza, and C. F. M. Toledo, "Harpia: A hybrid system for agricultural uav missions," *Smart Agricultural Technology*, vol. 4, p. 100191, 2023.

[243] A. Kuenz, J. Lieb, M. Rudolph, A. Volkert, D. Geister, N. Ammann, D. Zhukov, P. Feurich, J. Gonschorek, M. Gessner *et al.*, "Live trials of dynamic geo-fencing for the tactical avoidance of hazard areas," *IEEE Aerospace and Electronic Systems Magazine*, vol. 38, no. 5, pp. 60–71, 2023.

[244] A. Kamal, J. Vidaurri, and C. Rubio-Medrano, "No-fly-zone: Regulating drone fly-overs via government and user-controlled authorization zones," in *Proceedings of the Twenty-fourth International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing*, 2023, pp. 522–527.

[245] P. Wu, X. Yuan, Y. Hu, and A. Schmeink, "Trajectory and user assignment design for uav communication network with no-fly zone," *IEEE Transactions on Vehicular Technology*, 2024.

[246] H.-I. Lee, H.-S. Shin, and A. Tsourdos, "Uav collision avoidance considering no-fly-zones," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 14 748–14 753, 2020.

[247] W. Fei, Z. Xiaoping, Z. Zhou, and T. Yang, "Deep-reinforcement-learning-based uav autonomous navigation and collision avoidance in unknown environments," *Chinese Journal of Aeronautics*, vol. 37, no. 3, pp. 237–257, 2024.

[248] K. Heo, G. Park, and K. Lee, "On correcting errors in existing mathematical approaches for uav trajectory design considering no-fly-zones," *arXiv preprint arXiv:2308.13001*, 2023.

[249] Z. Lin, L. Castano, E. Mortimer, and H. Xu, "Fast 3d collision avoidance algorithm for fixed wing uas," *Journal of Intelligent & Robotic Systems*, vol. 97, no. 3, pp. 577–604, 2020.

[250] A. Waqas, D. Kang, and Y.-J. Cha, "Deep learning-based obstacle-avoiding autonomous uavs with fiducial marker-based localization for structural health monitoring," *Structural Health Monitoring*, vol. 23, no. 2, pp. 971–990, 2024.

[251] G.-T. Tu and J.-G. Juang, "Uav path planning and obstacle avoidance based on reinforcement learning in 3d environments," in *Actuators*, vol. 12, no. 2. MDPI, 2023, p. 57.

[252] S. Majumder, R. Shankar, and M. S. Prasad, "Obstacle size and proximity detection using stereo images for agile aerial robots," in *2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN)*. IEEE, 2015, pp. 437–442. [Online]. Available: https://ieeexplore.ieee.org/document/7094292

[253] M. Sharma, A. Gupta, S. K. Gupta, S. H. Alsamhi, and A. V. Shvetsov, "Survey on unmanned aerial vehicle for mars exploration: Deployment use case," *Drones*, vol. 6, no. 1, p. 4, 2021.

[254] T. Wu, Z. Zhang, F. Jing, and M. Gao, "A dynamic path planning method for uavs based on improved informed-rrt* fused dynamic windows," *Drones*, vol. 8, no. 10, p. 539, 2024.

[255] M. A. A. dos Santos and K. C. T. Vivaldini, "A review of the informative path planning, autonomous exploration and route planning using uav in environment monitoring," in *2022 International conference on computational science and computational intelligence (CSCI)*. IEEE, 2022, pp. 445–450.

[256] J. Rückin, L. Jin, and M. Popović, "Adaptive informative path planning using deep reinforcement learning for uav-based active sensing," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 4473–4479.

[257] J. Rückin, F. Magistri, C. Stachniss, and M. Popović, "An informative path planning framework for active learning in uav-based semantic mapping," *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4279–4296, 2023.

[258] M. O. Orisatoki, "Informative path planning for autonomous mapping of unknown areas," Ph.D. dissertation, University of Sussex, Apr. 2025. [Online]. Available: https://sussex.figshare.com/articles/thesis/Informative_path_planning_for_autonomous_mapping_of_unknown_areas/28730846

[259] N. B. Rossello, R. F. Carpio, A. Gasparri, and E. Garone, "Information-driven path planning for uav with limited autonomy in large-scale field monitoring," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 2450–2460, 2021.

[260] N. Michel, A. Patnaik, Z. Kong, and X. Lin, "Energy-optimal planning of waypoint-based uav missions - does minimum distance mean minimum energy?" in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 10 362–10 369.

[261] T. Czachórski, E. Gelenbe, G. S. Kuaban, and D. Marek, "Optimizing energy usage for an electric drone," in *EuroCybersec 2021, Communications in Computer and Information Science*, vol. 1596. Springer, 2022, pp. 61–75.

[262] J. Modares, F. Ghanei, N. Mastronarde, and K. Dantu, "Ub-anc planner: Energy efficient coverage path planning with multiple drones," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 6182–6189.

[263] L. Nam, L. Huang, X. J. Li, and J. Xu, "An approach for coverage path planning for uavs," in *2016 IEEE 14th international workshop on advanced motion control (AMC)*. IEEE, 2016, pp. 411–416.

[264] A. A. Meera, M. Popović, A. Millane, and R. Siegwart, "Obstacle-aware adaptive informative path planning for uav-based target search," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 718–724.

[265] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

[266] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Pearson Education, 2010.

[267] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: Optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 1, pp. 29–41, 1996.

[268] M. Dorigo and G. Di Caro, "Ant colony optimization: a new meta-heuristic," in *Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406)*, vol. 2. IEEE, 1999, pp. 1470–1477.

[269] C. B. Browne, E. Powley, D. Whitehouse, S. M. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton, "A survey of monte carlo tree search methods," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 4, no. 1, pp. 1–43, 2012.

[270] M. Popović, G. Hitz, J. Nieto, I. Sa, R. Siegwart, and E. Galceran, "Online informative path planning for active classification using uavs," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 5753–5758.

[271] Y. Xu, R. Zheng, S. Zhang, M. Liu, and J. Yu, "Online informative path planning of autonomous vehicles using kernel-based bayesian optimization," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 71, no. 8, pp. 3790–3794, 2024.

[272] H. Ergezer and K. Leblebicioglu, "Path planning for uavs for maximum information collection," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 1, pp. 502–520, 2013.

[273] DJI, "Dji matrice 300 rtk specs," 2023. [Online]. Available: https://www.dji.com/matrice-300/specs

[274] S. Ann, Y. Kim, and J. Ahn, "Area allocation algorithm for multiple uavs area coverage based on clustering and graph method," *IFAC-PapersOnLine*, vol. 48, no. 9, pp. 204–209, 2015.

[275] F. Balampanis, I. Maza, and A. Ollero, "Area partition for coastal regions with multiple uas," *Journal of Intelligent & Robotic Systems*, vol. 88, no. 2, pp. 751–766, 2017.

[276] J. J. Acevedo, I. Maza, A. Ollero, and B. C. Arrue, "An efficient distributed area division

method for cooperative monitoring applications with multiple uavs," *Sensors*, vol. 20, no. 12, p. 3448, 2020.

[277] P. L. K. Priyadarsini *et al.*, "Area partitioning by intelligent uavs for effective path planning using evolutionary algorithms," in *2021 International Conference on Computer Communication and Informatics (ICCCI)*. IEEE, 2021, pp. 1–6.

[278] Y. Chen, Z. Mou, B. Lin, T. Zhang, and F. Gao, "Complete coverage path planning for data collection with multiple uavs," in *2024 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2024, pp. 1–6.

[279] C. Zhang, C. Xu, X. Cheng, X. Li, G. Li, and B. He, "Multi-region joint coverage for environmental monitoring using energy-constrained uavs," *IEEE Transactions on Instrumentation and Measurement*, 2025.

[280] C. Gao, X. Wang, X. Chen, and B. M. Chen, "A hierarchical multi-uav cooperative framework for infrastructure inspection and reconstruction," *Control Theory and Technology*, vol. 22, no. 3, pp. 394–405, 2024.

[281] J. Gui, T. Yu, B. Deng, X. Zhu, and W. Yao, "Decentralized multi-uav cooperative exploration using dynamic centroid-based area partition," *Drones*, vol. 7, no. 6, p. 337, 2023.

[282] A. N. Jati, B. R. Trilaksono, E. M. I. Hidayat, and W. Adiprawita, "Collaborative coverage strategy using multiple uavs-ugvs in crn mapping," *IEEE Access*, 2025.

[283] X. Tian, M. Afrin, S. Mistry, R. Mahmud, A. Krishna, and Y. Li, "Mure: Multi-layer real-time livestock management architecture with unmanned aerial vehicles using deep reinforcement learning," *Future Generation Computer Systems*, vol. 161, pp. 454–466, 2024.

[284] G. Jocher and J. Qiu, "Ultralytics yolo11," 2024. [Online]. Available: https://github.com/ultralytics/ultralytics

[285] M. Lyu, Y. Zhao, C. Huang, and H. Huang, "Unmanned aerial vehicles for search and rescue: A survey," *Remote Sensing*, vol. 15, no. 13, p. 3266, 2023.

[286] P. Wang, C. Wang, J. Wang, and M. Q.-H. Meng, "Quadrotor autonomous landing on moving platform," *Procedia Computer Science*, vol. 209, pp. 40–49, 2022.

[287] J. Guérin, K. Delmas, and J. Guiochet, "Certifying emergency landing for safe urban uav," in *2021 51st Annual IEEE/IFIP international conference on dependable systems and networks workshops (DSN-W)*. IEEE, 2021, pp. 55–62.

[288] A. Rodriguez-Ramos, C. Sampedro, H. Bavle, P. De La Puente, and P. Campoy, "A deep reinforcement learning strategy for uav autonomous landing on a moving platform," *Journal of Intelligent & Robotic Systems*, vol. 93, pp. 351–366, 2019.

[289] D. Falanga, A. Zanchettin, A. Simovic, J. Delmerico, and D. Scaramuzza, "Vision-based autonomous quadrotor landing on a moving platform," in *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*. IEEE, 2017, pp. 200–207.

[290] A. Tullu, M. Hassanalian, and H.-Y. Hwang, "Design and implementation of sensor platform for uav-based target tracking and obstacle avoidance," *Drones*, vol. 6, no. 4, p. 89, 2022.

[291] H. Tovanche-Picon, J. González-Trejo, Á. Flores-Abad, M. Á. García-Terán, and D. Mercado-Ravell, "Real-time safe validation of autonomous landing in populated areas: from virtual environments to robot-in-the-loop," *Virtual reality*, vol. 28, no. 1, p. 66, 2024.

[292] G. Jocher, "Yolov5 by ultralytics," 2020. [Online]. Available: https://github.com/ultralytics/yolov5

[293] J. Y. Lee, A. Y. Chung, H. Shim, C. Joe, S. Park, and H. Kim, "Uav flight and landing guidance system for emergency situations," *Sensors*, vol. 19, no. 20, p. 4468, 2019.

[294] X. Zhou, Z. Wang, H. Ye, C. Xu, and F. Gao, "Ego-planner: An esdf-free gradient-based local planner for quadrotors," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 478–485, 2020.

[295] C. M. Belcastro, R. L. Newman, J. Evans, D. H. Klyde, L. C. Barr, and E. Ancel, "Hazards identification and analysis for unmanned aircraft system operations," in *17th AIAA Aviation Technology, Integration, and Operations Conference*, 2017, p. 3269.

[296] M. W. Mueller, M. Hehn, and R. D'Andrea, "A computationally efficient motion primitive for quadrocopter trajectory generation," *IEEE transactions on robotics*, vol. 31, no. 6, pp. 1294–1310, 2015.

[297] S. Magnusson and P. P. Hagerfors, "Drone deliveries of medical goods in urban healthcare," Master's thesis, Chalmers University of Technology, Gothenburg, Sweden, 2019. [Online]. Available: https://www.vgregion.se/contentassets/74ed4a24d73c4f648bcc87aa22721539/drone-delivieries-of-medical-goods-in-urban-healthcare---final.pdf

[298] I. Quads, "Iq tutorials," Online, 2024, available: https://github.com/Intelligent-Quads/iq_tutorials.

[299] A. D. Team, "Ardupilot: Open source autopilot system," Online, 2024, available: https://ardupilot.org/.

[300] M. Developers, "Mavros: Mavlink extendable communication node," Online, 2024, available: https://github.com/mavlink/mavros.