



FACULTY OF ELECTRICAL ENGINEERING
DEPARTMENT OF CYBERNETICS
MULTI-ROBOT SYSTEMS

ROBUST UAV LOCALIZATION IN PERCEPTION-DEGRADED ENVIRONMENTS

DOCTORAL THESIS

ING. PAVEL PETRÁČEK

DOCTORAL PROGRAM
INFORMATICS

SUPERVISOR
DOC. ING. MARTIN SASKA, DR. RER. NAT.

IN PRAGUE ON MARCH 28, 2024

Acknowledgments

I express my gratitude to Martin Saska for his compassionate supervision throughout my Ph.D. studies. I appreciate Martin's relaxed leadership and the invaluable motivation he provided to inspire my doctoral studies. I am incredibly grateful for the freedom of research afforded to me in the final year of my studies, and I value the myriad of academic and managerial lessons shaping not only my research but also my professional growth.

I thank the members of the Multi-Robot Systems (MRS) laboratory for our collective endeavor in pushing the boundaries of aerial robotics. The friendships forged during our studies have been instrumental in my journey on both academic and personal levels. Each of you has taught me important life lessons and influenced me for the better. I take immense pride in our collaborative achievements and am grateful for the privilege of being part of MRS. Specifically, I would like to express my gratitude to Vít Krátký for our fruitful, complementary, and balanced efforts not only during sleepless nights when writing papers and experimenting out of home, Matěj Petrlík and Jan Bednář for partaking in plentiful hobbies as well as their unwavering camaraderie and endless assistance whenever it was needed the most, Tomáš Báča and Daniel Heřt for sharing their boundless knowledge and talents far exceeding the field of robotics, and Vojtěch Vonásek for showing me that passing knowledge can be gratifying. Tomáš, Matěj, Daniel, and Vojtěch Spurný also earn a special mention for their foundational work on our research system, which was used daily throughout my research endeavors. Although I cannot list all the members of MRS, I thank my past and present colleagues no less for your work. I will forever remember and benefit from our hard-worked experimental campaigns, supervising drones in a church illuminated only by an outside thunderstorm, and sending a plethora of robots beyond a point of return. This thesis would not have existed without your work, assistance, solidarity, and contributions, and I have cherished every moment of the jerky flight it has been.

I would also like to acknowledge the members of our Dronument project, of which I am very proud. I thank Milan Škobrtal from the National Heritage Institute of the Czech Republic for collaborating and administering the historical monuments where we could push the boundaries of robotics. And once more, I thank Vít Krátký for sharing the front seats of stress, responsibility, and rewards of such an exciting adventure. I also thank all the members of team CTU-CRAS-NORLAB participating in SubT, with whom I was privileged to share the fun and sport of the competition.

I am also grateful to the administrative staff and essential workers at the Department of Cybernetics at FEE CTU, where I started as a young bachelor's student in 2015. Although my gratitude goes to the administration, secretaries, the IT department, cleaning staff, and guards, I want to specifically thank Kristina Lukešová for her assistance with every administrative problem there ever was. Your consistent efforts have greatly contributed to streamlining my day-to-day operation.

I extend my heartfelt thanks to my family and friends for their continuous support throughout the journey of life. Last but certainly not least, I am profoundly grateful to my partner, Veronika, for her emotional support and for providing much-needed distractions from my studies. Her presence in my life has been invaluable in completing my doctoral studies while maintaining my sanity. I am eternally grateful for the immense joy and comfort she brings to my life.

Wholeheartedly, thank you.



Figure 1: My colleagues from the Multi-Robot Systems laboratory at Faculty of Electrical Engineering (FEE), Czech Technical University (CTU) in Prague.

Funding

During my Ph.D. studies, I have been supported by the taxpayers of the Czech Republic through a Ph.D. scholarship. I was also supported by research projects financed by multiple institutions and agencies, namely

- by the Czech Technical University in Prague through grants SGS17/187/OHK3/3T/13, SGS20/174/OHK3/3T/13, and SGS23/177/OHK3/3T/13,
- by the Czech Science Foundation under research projects no. 17-16900Y, 20-29531S, 20-10280S, 22-24425S, 23-06162M,
- by the Ministry of Education of the Czech Republic in OP VVV funded project “Research Center for Informatics” no. CZ.02.1.01/0.0/0.0/16 019/0000765,
- by the Ministry of Culture of the Czech Republic under project no. DG18P02OVV069 in program NAKI II,
- by the European Union in Horizon 2020 research and innovation program AERIAL-CORE under grant agreement no. 871479 and under the project Robotics and Advanced Industrial Production (reg. no. CZ.02.01.01/00/22_008/0004590),
- by the Technology Agency of the Czech Republic (TAČR) under projects no. FW03010020 and FW01010317,
- by the US Defense Advance Research Projects Agency,
- by the Technology Innovation Institute — Sole Proprietorship LLC, UAE under the Research Project Contract No. TII/ATM/2032/2020, and
- by the project “e-Infrastruktura CZ” (e-INFRA LM2018140) provided within the program Projects of Large Research, Development and Innovations Infrastructures, which supplied computational resources.

Copyright

This thesis is a compilation of several articles published and submitted during the authors' Ph.D. studies. The included publications are presented under the copyrights of IEEE and IOP Publishing for posting the works for internal institutional uses. The works are protected by the copyrights of respective publishers and can not be further reprinted without the publishers' permission.

© IEEE, 2020, 2021, 2023

© IOP Publishing, 2020

Abstract

This thesis delves into the problems related to the lack of robustness in the perception of autonomous mobile robots, focusing on the advancement of perceptual resiliency under diverse sources of degradations in GNSS-restricted environments and the applicability on-board robots with limited computational capabilities, such as multi-rotor Unmanned Aerial Vehicles (UAVs). The lack of resiliency of perception and autonomy is an ongoing challenge that hinders the use of autonomous robots in extreme environments containing geometrical symmetry, featureless structures, variable lighting, fog, and whirling dust, as well as many other sources of perceptual degradation. Devising robustness to extreme conditions is of the utmost importance for the applicability of autonomous robots in the unforgiving conditions of the real world, in which autonomous robots yield immense potential not only in assisting human personnel in repetitive and hazardous tasks. The threefold objectives of this thesis lie in I) advancing the perceptual capabilities of resource-constrained robots in degraded environments, II) distributing perceptual capabilities among the members of a multi-robot team, and III) transferring theoretical advancements beyond the laboratory. The first objective is contributed to by novel methods in LiDAR-based perception. These methods include maximizing observability in localization, redundancy-minimizing sampling of LiDAR data for real-time pose estimation, and robust dust filtering in extreme conditions. Contributions to the second objective include novel models for decentralized perception-aware swarming of communication-free UAV teams. The theoretical contributions and interdisciplinary branches culminate in real-world settings, focusing on the multi-robot tasks of search and rescue in the DARPA Subterranean Challenge (SubT), as well as in the inspection and documentation of the priceless interiors of historical monuments in the Dronument project. With the former, these contributions were integral to the UAV stack that proved to be the best-performing and most resilient aerial autonomy deployed in the harsh and diversified conditions of the renowned DARPA SubT competition. The latter transfers these contributions to the world-unique utilization of multi-UAV teams in the interiors of eighteen historical structures, including two sites on the UNESCO World Heritage list. This thesis is a compilation of six journal publications, each contributing to the defined objectives and focusing on the complex aspects of robust real-world perception and autonomy of UAV teams.

Keywords Mobile Robotics · Localization · Perception · Perceptual Degeneracy · Unmanned Aerial Vehicles · Multi-UAV Teams · Autonomy in Real-World · Aerial Swarming · Operational Research · DARPA Subterranean Challenge · Dronument

Abstrakt

Tato práce se zabývá problémy nedostatečné robustnosti vnímání autonomních a mobilních robotů a zaměřuje se na výzkum odolnosti robotické percepce pod vlivem percepční degradace v prostředích bez přístupu ke GNSS a aplikovatelnosti na palubě robotů s omezenými výpočetními zdroji, jako jsou vícerotorové bezpilotní prostředky (UAVs). Nedostatečná odolnost robotické percepce a autonomie je stávajícím výzkumným problémem, který brání využití autonomních robotů v extrémních prostředích, jenž jsou geometricky symetrické a obsahují bezpříznakové struktury, proměnlivé osvětlení, mlhu a vířící prach a mnoho dalších zdrojů percepční degradace. Výzkum odolnosti vůči extrémním podmínkám je nesmírně důležitý pro aplikovatelnost autonomních robotů v nelítostných podmínkách reálného světa, ve kterém autonomní roboti nabízejí obrovský potenciál ne jenom při asistenci lidskému personálu v často opakovaných a nebezpečných úlohách. Tři hlavní cíle této práce spočívají v I) pokroku percepce v degradovaných prostředích u robotů s omezenými výpočetními zdroji, II) distribuci percepčních schopností mezi členy týmu vícera robotů a III) převodu základního výzkumu mimo laboratoř. Mezi výstupy prvního cíle tato práce přispívá novými percepčními metodami využívajícími laserové senzory. Tyto metody zahrnují lokalizaci s maximální pozorovatelností, minimalizaci redundance laserových dat pro odhad polohy v reálném čase a robustní filtrování prachu z laserových dat v extrémních podmínkách. Výstupy druhého cíle zahrnují nové modely pro decentralizované roje UAV, jež se koordinují bez komunikace pouze na základě palubní percepce. Teoretické přínosy a interdisciplinární přesahy této práce jsou posunuty do podmínek reálného světa, kde se zaměřují na vícerobotické úlohy search and rescue (S&R) v DARPA Subterranean Challenge (SubT) a na inspekci a dokumentaci neocenitelných interiérů historických památek v projektu Dronument. V úlohách S&R byly příspěvky práce nedílnou součástí palubního systému bezpilotních UAV, který se ukázal jako nejlépe fungující a nejodolnější leteckou autonomií nasazenou v drsných a různorodých podmínkách prestižní soutěže DARPA SubT. V úlohách projektu Dronument jsou přínosy této práce aplikovány do světově unikátního využití týmů autonomních UAV v interiérech osmnácti historických památek, z nichž dvě jsou zařazeny na seznamu UNESCO. Tato práce je složena z šesti článků publikovaných v odborných časopisech, z nichž každý přispívá k definovaným cílům a zaměřuje se na komplexní aspekty robustní percepce a autonomie týmu bezpilotních prostředků v reálném světě.

Klíčová slova Mobilní Robotika · Lokalizace · Percepce · Percepční Degenerace · Bepilotní Letecké Prostředky · Týmy Bepilotních Prostředků · Autonomie Robotů v Reálném Světě · Roje Bepilotních Prostředků · Operační Výzkum · DARPA Subterranean Challenge · Dronument

Abbreviations

▪ CTU	Czech Technical University
▪ FEE	Faculty of Electrical Engineering
▪ CRAS	Center for Robotics and Autonomous Systems
▪ MRS	Multi-Robot Systems
▪ DARPA	Defense Advance Research Projects Agency
▪ SubT	Subterranean Challenge
▪ UVDAR	Ultraviolet Detection and Ranging
▪ MBZIRC	Mohamed Bin Zayed International Robotics Challenge
▪ LOAM	Lidar Odometry and Mapping in Real-time
▪ LM	Levenberg–Marquardt
▪ GTSAM	Georgia Tech Smoothing and Mapping
▪ PGO	pose graph optimization
▪ BA	bundle adjustment
▪ KLD	Kullback-Leibler divergence
▪ RTI	reflectance transformation imaging
▪ SLAM	simultaneous localization and mapping
▪ S&R	search and rescue
▪ GNSS	Global Navigation Satellite System
▪ UAV	Unmanned Aerial Vehicle
▪ MOCAP	motion capture
▪ VIO	visual-inertial odometry
▪ IR	infrared
▪ CNN	convolutional neural network
▪ LiDAR	light detection and ranging
▪ V-SLAM	visual SLAM
▪ L-SLAM	LiDAR SLAM
▪ LI	LiDAR-inertial
▪ VI	visual-inertial
▪ LVI	LiDAR-visual-inertial
▪ ICP	iterative closest point
▪ IEKF	iterated extended Kalman filter
▪ ESKF	error-state Kalman filter
▪ NDT	normal distributions transform
▪ FoV	field-of-view
▪ UWB	ultra-wideband
▪ DoF	degree-of-freedom
▪ IMU	inertial measurement unit
▪ WoS	Web of Science
▪ GS	Google Scholar
▪ JIF	Journal Impact Factor
▪ SF	self-citation
▪ SCIE	Science Citation Index Expanded
▪ ESCI	Emerging Sources Citation Index
▪ CRedit	Contributor Roles Taxonomy
▪ RAS	Robotics and Automation Society

Contents

1	Introduction	1
1.1	Thesis Organization	5
2	Contributions and Related Work	6
2.1	LiDAR-based Robot Perception	6
2.2	Distributed Multi-Robot Sensing and Acting	14
2.3	Theory to Practice: Validation of Fundamental Robotic Research	17
3	LiDAR-based Perception in Demanding Real-World Conditions	19
4	Distributed Perception-Aware Multi-Robot Coordination	124
5	Discussion and Summary of Results	153
5.1	LiDAR-based Perception and UAV Autonomy in Demanding Real-World Con- ditions	153
5.2	Distributed Perception-Aware Multi-UAV Coordination	156
5.3	Highlights, Interdisciplinary Results, and Secondary Activities	156
6	Conclusion	159
A	Author’s List of Publications	160
A.1	Core peer-reviewed publications	160
A.2	Other peer-reviewed publications	161
A.3	Thesis-unrelated publications	164
B	Citations of Authors’ Publications	165
B.1	List of Citations	165
C	References	170

■ 1 Introduction

The research field of mobile robotics has reached a stage where autonomous mobile robots are being deployed to a limited extent in applications within the industrial sector, namely for automation, logistics, and periodic inspection. However, this trend predominantly applies to ground-based mobile robots (e.g., ANYmal by [®]ANYbotics [20]), which are subject to relatively nominal safety requirements during interactions with both their environment and nearby humans. In ground-based robotics, the safety requirements can be relaxed due to the feasibility of deactivating the robots quickly and safely in the event of a malfunction. In contrast, the safety requirements for **autonomous aerial robots** are significantly more strict due to potential risks to human personnel, valuable assets in the environment, and the robots themselves. Such demands on aerial-robotic safety hinder the current applicability of UAVs, as theoretical and algorithmic challenges in resiliency and reliability in their autonomous functions persist, especially when operated indoors without access to global systems.

The primary motivation of this thesis delves into the problems of insufficient robustness, reliability, and resiliency of resource-constrained aerial robots and teams containing them, especially in settings devoid of Global Navigation Satellite System (GNSS) availability. This thesis explores the obstacles to achieving reliable and replicable UAV autonomy and presents the author’s contributions in overcoming them, from enhancing robot localization in perceptually-aliased environments, to advancing multi-robot decision-making in minimum-information scenarios. All presented contributions are based on the single **premise** — research methods that yield minimal operational latency and are capable of operating in real-time and on-board robots with constrained computational resources. These aspects are particularly crucial for small-factor multi-rotor UAVs with inherently unstable dynamics, where minimizing the size and weight without compromising computational efficiency and real-time properties is paramount, especially for rapid and agile 3D deployment. This premise builds upon challenges defined first in [21] and confirmed as an ongoing challenge in the recent work [7a] (co-authored by the author of this thesis together with authors from most of the leading laboratories in the field), which builds upon the experience from the extreme environments in the SubT robotic competition organized by the Defense Advance Research Projects Agency (DARPA). Both works [7a, 21], authored by leading experts in field robotics, specify the need for developing perceptual systems robust towards perceptual degeneracies (such as geometrical symmetry and variable lighting) and capable of operating onboard resource-constrained robots. Given the premise, we specify below the primary challenges related to the motivations of the thesis and define the research objectives for tackling them.

Challenge (1): fast and resilient onboard perception. Since a robot’s ability to perceive and act within its environment is fundamental to mobile robotics, challenges in these areas significantly impede the broader adoption of autonomous UAVs. A moderately optimistic perspective on the current state-of-the-art robotics might mistakenly assume that challenges in multi-rotor UAV stabilization and control, state estimation, localization, mapping, navigation, motion planning, and communication have been solved, thereby enabling full on-board deployment even in environments devoid of GNSS. This assumption might be valid to some extent, but only under the assumption of a perfectly functioning, precise, non-drifting, real-time, and resilient on-board perception of the robot. The robot’s perception precedes all reliant modules of its autonomy, ranging from state estimation to control, motion planning, navigation, mapping, and decision making. This makes the perception, alongside state estima-

tion, the critical bottleneck of aerial autonomy, especially in GNSS-denied environments. Nevertheless, this misconception emerges from recent advances in sensory and computational capabilities, leading to remarkable levels of progress in perceptual systems over the past decade. Under optimal conditions, robots are capable of accurately localizing themselves in a dynamic environment [7a, 21, 22], mapping their surroundings (e.g., topologically [23, 24], semantically [25, 26], volumetrically [27, 28]), detecting and tracking moving objects [29, 30], and generally performing tasks, that would be classified as science fiction just a few decades ago. Yet, these capabilities are often computationally expensive and vulnerable to various perceptual degradations, which yield an ongoing challenge for the resilience and robustness of robotic systems [7a]. The causes and effects of these perceptual degradations are linked to multiple factors, including noise in LiDAR-based and vision-based perception in dusty environments or the emergence of drift in pose estimation of LiDAR-based simultaneous localization and mapping (SLAM) in geometrically symmetrical environments, as well as in dark and texture-less environments for systems reliant on visual information. This thesis addresses the issues of perceptual degradation and aliasing, highlighting their implications for UAV autonomy [1c–4c, 6c].

Objective (1): advance LiDAR-based methodology tackling perceptual degeneracy in GNSS-restricted and geometrically symmetrical environments. Many natural and man-made environments are geometrically structureless and symmetrical. Such environments contain a few geometrical landmarks or features, which can be exploited by onboard perception. A lack of geometrical features makes it difficult to distinguish, if even possible, a robot pose change from on-board LiDAR data. Note that in visual methods, featureless images (such as white planes in urban areas) are equivalent to geometrical non-structurality in LiDAR-based methods. If a robot obtains two identical measurements at two different places in an environment, the perception is denoted as aliased and the environment degenerate. Examples of degenerate environments include long narrow tunnels (e.g., metros), urban or industrial corridors, and large historic buildings (e.g., cathedrals and churches), which are typically unstructured along the vertical axis (see Figure 1.1). Given our premise, this objective challenges the problems of robustness and scalability of SLAMs defined in [21] — the vulnerability of optimization solvers in the presence of outliers, lack of awareness of imminent failures due to perceptual degeneracies, and adaptation of the methodologies to resource-constrained platforms. In other words, the problems include the rejection of spurious features and their accurate temporal and spatial association as well as detection, prediction, and minimization of degraded perception on-board lightweight platforms, such as small-scale UAVs. Our contributions to this objective are proposed in core publications in Chapter 3 and are summarized in Section 2.1.

Challenge (2): safe and reliant multi-robot coordination. State-of-the-art research in S&R [3c, 4a, 7a, 8, 31], the documentation of historical buildings [1c, 2a, 9a, 10a], in industrial inspection [32, 33], cinematographic filming [34, 35], and many other robotic subfields utilize teams of cooperating autonomous robots in their respective tasks. Apart from the bottleneck of non-resilient perception, the robots cooperating within compact teams usually rely on a centralization element, global positioning, and continuous communication in order to sense and act within their shared environment. However, these design choices serve as critical points of failure and introduce major vulnerabilities into the multi-robot scheme. A methodology proposing to remove these critical points of possible failures emerges in the field of decentralized perception-aware swarming. In this field, the decentralized architecture follows the definition of an autonomous swarm as defined in [36] — a multi-agent team that is scalable to large groups, has high redundancy and fault tolerance, is usable in tasks unsolvable by a single robot, and has locally limited sensing. In contrast to typical architectures, a multi-robot system satisfying the definition of [36] is robust to the single points of failure, which makes it resilient to

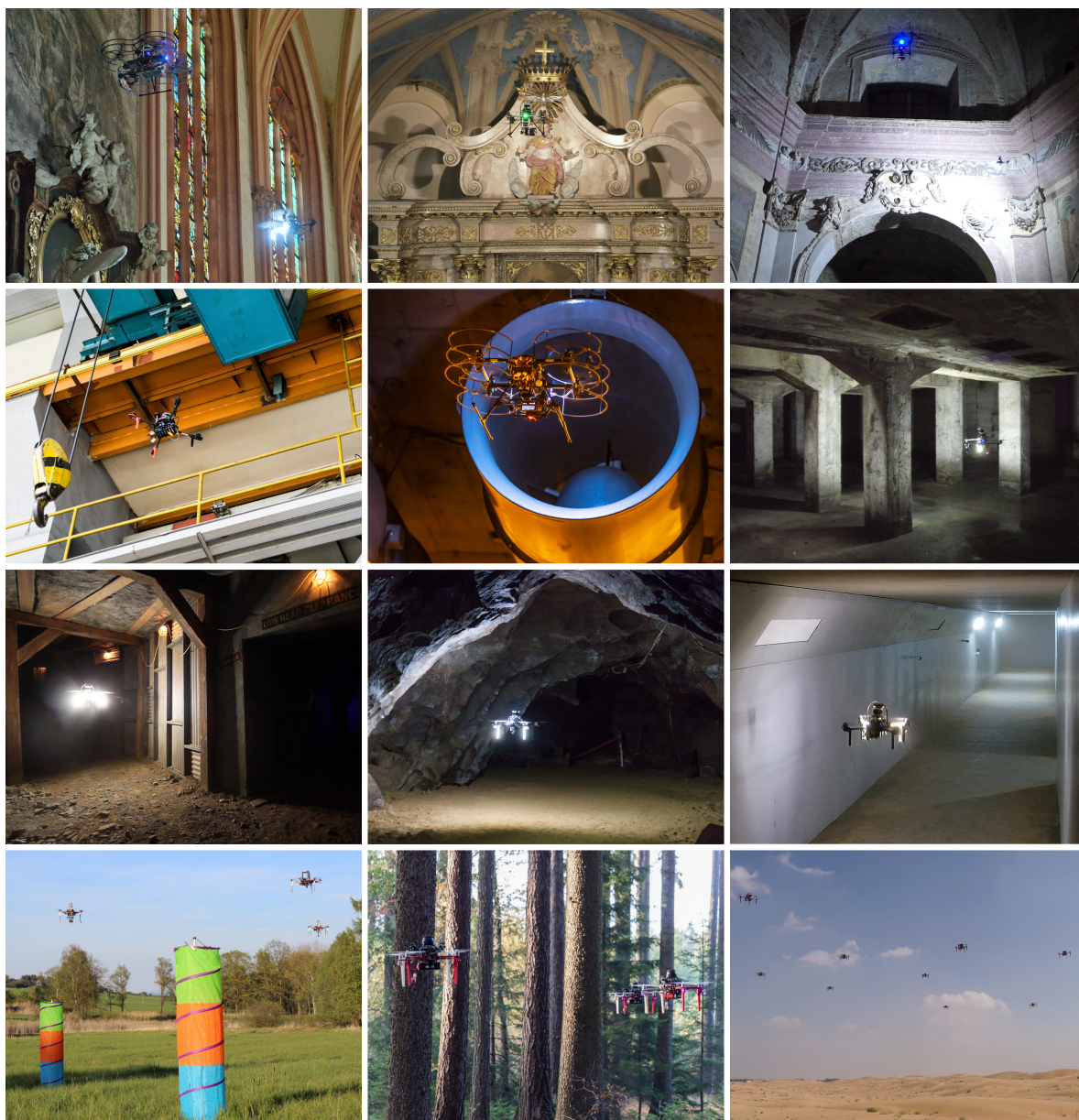


Figure 1.1: Demonstrations of the author’s theoretical contributions as actualized in the perceptually-challenging conditions of real-world 3D space. The selected showcases relate to fully autonomous multi-rotor UAV (or teams of such) deployments in tasks of inspecting and documenting historical and industrial interiors, exploring diverse and demanding subterranean environments, and perception-aware swarming in communication-less settings.

the common bottlenecks of the current state of the art, especially in relation to tightly cooperating multi-UAV teams. In other words, substituting the critical system-design characteristics (i.e., centralization, global perception) with decentralization and information retrieval from direct perception improves the system’s overall reliability and increases its scalability bound.

Objective (2): research distributed multi-robot sensing and acting for the communication-free coordination of a multi-UAV team employing direct relative localization among its members. Compact multi-robot teams require synchronized and accurate decision

making in order to coordinate safely and efficiently. Removing the centralization and communication elements makes such coordination challenging. Nevertheless, safety and independence on the single points of failure in cooperative tasks can be achieved with decentralized swarming models designed for sensing and acting on the basis of local information. Despite such models being studied in relation to modeling behavior of biological systems (such as flocks of birds or schools of fish), we propose to investigate their potential in serving as a fast, drift-free, perception-aware, and communication-independent navigation of a multi-UAV team. Such investigations shall study the effects of retrieving local information from onboard perception, especially in environments filled with obstacles and in scenarios where global systems (such as localization) and multi-robot communication are not available or are subject to malfunction. The research objective needs to investigate and integrate novel methods for relative localization of nearby robots (such as Ultraviolet Detection and Ranging (UVDAR) [37, 38]) as a mechanism for local-information retrieval in multi-robot systems. With the primary objective being the maximization of robustness in multi-robot systems, the safety and resiliency margins need to be derived from theoretical and experimental analyses, especially with respect to the errors and delays in the perceptual systems of individual robots cooperating within a compact team moving in 3D space. The author’s contributions to this objective are proposed in core publications in Chapter 4 and are summarized in Section 2.2.

Challenge (3): advancing fundamental research outside the laboratory. Theoretical contributions in modern robotic research require thorough experimental validations in challenging real-world conditions to objectively validate their applicability. This is especially important in mobile robotics, where theoretical hypotheses and their evaluation and comparison in simulation, laboratory, and offline datasets are seldom enough. Thus, thorough analyses in real-world scenarios are crucial, particularly in using affordable hardware onboard the robots as well. Real-world experimental analyses provide valuable information about the deployability of the robotic methodology and support theoretical contributions proposed in a robotic work. This is rooted in the fact that many simplifying assumptions do not hold in the real world, including hard-to-model constraints composed of sensory noise, inaccuracies, approximations, and time delays, among other factors. When coupled with other onboard systems, the applicability of theoretical contributions may consequently prove to be limited. The critical need for field testing is visible in the amount of robotic competitions funded by private and public organizations with the common goal of pushing the state of the art in field robotics and real-world deployment of autonomous robots. Among the most notable competitions are Mohamed Bin Zayed International Robotics Challenge (MBZIRC), which challenges aerial robotics in diverse tasks requiring precise coordination of robots, and the DARPA SubT. The challenges of DARPA SubT ranged from resilient perception to on-the-run mapping, decision-making, and cooperation of a heterogeneous team, all inside closely designed subterranean environments, such as man-made mines, natural caves, and urban structures. In addition, multiple past and ongoing research projects in aerial robotics (such as multi-disciplinary documentation of historical interiors by multi-UAV teams [2c]) have shown that even modern contributions require further fundamental research in obtaining sufficient aerial reliability in real-world conditions.

Objective (3): transfer of fundamental research of (1) and (2) to real-world tasks, including single UAV deployments and multi-robot tasks that can not principally be performed with a single UAV. The objective aims to include validation of the theoretical contributions in diversified scenarios of the real 3D world, verifying the direct applicability of the research. This objective is connected with the multi-disciplinary motivation of utilizing fast, resilient, and repeatable autonomy in challenges requiring distributed coordination, primarily for the purposes of maximum-information coverage [4c], efficient environment exploration [8a],

and cooperative documentation using heterogeneous robots [2c]. This objective aims to push the state of the art in mobile robotics by applying the results of theoretical research in variable real-world tasks that require UAVs of minimal size with limited computational and sensory resources on board. This objective builds upon the common avenue of the MRS laboratory at CTU and its MRS UAV System [39], which provides a modular framework for controlling and deploying multi-rotor UAVs in both virtual and real environments. Selected illustrations of our contributions to this objective are showcased in Figure 1.1. The list of objective-related contributions, especially in S&R and interior documentation by autonomous UAV teams, is summarized in Section 2.3 and is part of our core publications described in Chapter 3 and Chapter 4.

■ 1.1 Thesis Organization

This thesis is a compilation of 6 included core publications, referenced as [*c]. However, the thesis is supported by an additional 13 authored and co-authored publications, referenced as [*a]. The remainder of this thesis is organized as follows.

- Chapter 2 describes the current state of the art related to the research objectives and incorporates a short summary of the author’s contributions in the field,
- Chapter 3 introduces the publications related to objectives (1) and (3) on developing LiDAR-based methods for resource-constrained UAVs in environments containing perceptual degeneracy,
- Chapter 4 presents the publications related to the objectives (2) and (3) on distributed sensing and acting of multi-agent teams,
- Chapter 5 discusses the achieved results,
- Chapter 6 concludes the thesis,
- Appendix A lists all of the author’s publications, including the author’s contributions and the credit author statement, and
- Appendix B lists the citations referencing the author’s publications.

■ 2 Contributions and Related Work

This chapter summarizes the relevant state of the art in mobile robotics and its relation to the author’s contributions. The summary includes the contributions presented within the core publications [1c–6c] (six journal publications), which comprise the core of this thesis, as well as contributions of other author’s research works [7a–19a] (eight impacted journal publications, three conference proceedings in Web of Science, and two secondary research activities), which extend the spectra of contributions. We refer to and describe the core publications individually in Chapter 3 and Chapter 4. Note that, with respect to the premise defined in Chapter 1, the goal of the contributions is to enhance the robustness, reliability, and safety of aerial systems with limited computation and communication resources on board. This includes removing points of possible system failures and designing lightweight solutions minimizing the computational latency and degradation of perception-based localization.

This chapter is divided into three sections, each relating to one of the challenges and objectives listed in Chapter 1. Section 2.1 describes the current state of LiDAR-based methods in robot state estimation while breaking it down into the subtopics of perceptual degradation, multi-modality, and point cloud sampling. Section 2.2 summarizes the state of the art in aerial swarming and focuses on distributed architectures, independence in communication, and the adoption of mutual perception among UAVs within a swarm. Section 2.3 briefly discusses the state of field robotics and summarizes the author’s contributions to challenge (3) and objective (3) on robotic experimentation.

■ 2.1 LiDAR-based Robot Perception

The first set of contributions of this thesis focuses on the LiDAR-based perception of an autonomous robot for the purposes of localization, odometry, and SLAM. In localization, a robot estimates its state given a map of the environment. In SLAM, however, the objective is to localize the robot in a priori unknown environment, while simultaneously building a map of the environment using an available sensory apparatus onboard the robot and passive or active landmarks present in the robot’s observable proximity. Furthermore, we use the term odometry as a subfield of SLAM in which loop closing (i.e., global map and trajectory optimization when revisiting known places) is omitted. Since the research field in this area is broad¹, the following discussion covers the methods compliant with the premise of this thesis. The focus hereafter is simultaneously put on LiDAR-based methodology and the challenges of ill-defined optimization problems, perception degradation, and lightweight solutions for resource-constrained robots.

Due to the recent technological advance in modern LiDAR technology (such as Velodyne Puck, which launched in the public market in 2014), the research focus of LiDAR-based SLAM is greatly on the rise in the last decade. The related literature within this field has since converged to utilizing three primary correlating techniques for reasoning over LiDAR data — scan matching, filtering for estimation, and graph-based optimization. The scan matching relies on scan-to-scan and/or scan-to-map registration, either in the point cloud space or in a feature space, with the latter being the most frequent and, herein, the primary scenario discussed (unless specified otherwise). The graph-based optimization (also referred to in the literature as bundle adjustment (BA)) is utilized in most modern approaches on both the local and

¹Tens of related papers are published every year in the most impacted journals and conference proceedings.

the global scale to solve the problem commonly classified as pose graph optimization (PGO). This optimization benefits from the concepts of pose-graph or factor-graph representations of the problem. It primarily utilizes one of the two main optimization frameworks in the field — g^2o [40] for pose-graph and Georgia Tech Smoothing and Mapping (GTSAM) [41] for factor-graph optimization. Although some state-of-the-art methods do not employ filtering for state estimation and rather substitute the filtering by creating recurrent factors during factor-graph-based optimization, the filtering methods are still commonly applied when working with asynchronous data sources with differing uncertainty. Nevertheless, most of the current state-of-the-art methods contain multifold sequential and/or parallel processing layers that often apply all three techniques to reduce the cumulative error of state estimation in the SLAM problem.

■ 2.1.1 LiDAR-based Localization, SLAM, and Odometry

The majority of state-of-the-art LiDAR SLAMs (L-SLAMs) are related to the methodology proposed in Lidar Odometry and Mapping in Real-time (LOAM) [42]. The authors of LOAM [42] proposed extracting plane and edge features from raw 3D LiDAR data, followed by a two-step optimization process working within the feature space. These two steps include fast local odometry that utilizes scan-to-scan matching and slow mapping employing local scan-to-map optimization. Both optimization processes match salient feature points using a non-linear iterative closest point (ICP) algorithm, while considering the intrinsic motion of a fast-rotating LiDAR by compensating for the single-axis rotation with a constant angular velocity model. Although LOAM does not provide global optimization in the form of loop detection and loop closure, the method has been shown to yield minimum drift on the KITTI dataset [43], where it still holds first place² among LiDAR-based methods. Highlighted among the LOAM-inspired methods (i.e., methods employing the LOAM feature extraction) is LeGO-LOAM [44], which is designed for use on-board ground vehicles by utilizing ground segmentation during the proposed two-step Levenberg–Marquardt (LM) optimization. The LeGO-LOAM method was further extended with scan context to LeGO-LOAM-SC [45], applying global descriptors extracted by scan context for loop detections. The scan context is an algorithm utilizing global descriptors of non-histograms to enable searching for correlations between current and previous LiDAR scans [46]. Another LOAM-inspired SLAM designed for ground vehicles was proposed in [47], where the authors introduced global optimization using ground-plane constraints, non-ground scan matching, and SegMatch-based loop detection based on matching segmented clusters of points. Local BA on a sliding window of keyframes was applied to LOAM in BALM [48], where the authors proposed to overcome the feature sparsity by formulating the BA on a sparse feature map and directly minimizing the distances of the feature points to entire edges or planes, and not to other feature points directly. The sliding window of keyframes is an effective method of visual SLAMs (V-SLAMs) not commonly applied to V-SLAMs as sparse feature points make the exact point matching challenging. Utilizing the information about the intensity of light beams reflected back to a LiDAR was proposed in Intensity-SLAM [49], which supplies geometric features with an intensity map and intensity residuals to increase pose estimation accuracy. The improvements to the odometry stage of the two-stage LOAM optimization were introduced in [50], where the authors proposed processing individual packets of the LiDAR data stream instead of the full-revolution scan in order to minimize delays and lower the odometry drift. A LiDAR-inertial (LI) odometry FAST-LIO [51] improves the accuracy of the odometry stage of LOAM by fusing the inertial measurements and the fast-odometry feature-matching optimization utilizing iterated extended Kalman filter (IEKF).

²Ranking available at cvlibs.net/datasets/kitti/eval_odometry.php (accessed on March 27, 2024).

Among the state-of-the-art L-SLAMs not employing the geometric features of LOAM are the stand out methods of KISS-ICP [52], FAST-LIO2 [53], Point-LIO [54], and SuMa [55] (with its improved version SuMa++ [56]). KISS-ICP [52] is a modern and parallelized implementation of the original ICP algorithm minimizing the point-to-point metric. FAST-LIO2 [53] and Point-LIO [54] directly register the 3D points while employing a kinematic model for propagating each point via IEKF using integrated inertial measurements. This per-point forward approach allows for fast and agile 6-DoF estimation. In contrast to other methods, it is suitable for sensors that retrieve a single point or irregular batches of points at a time. The SuMa methods [55, 56] utilize surfel-based mapping with integrated semantic information extracted by a deep-learning convolutional neural network (CNN) in semantic ICP. Another highly efficient method, LiTAMIN2 [57], applies geometric surfel-based approximation to reduce the number of points, thereby speeding up the scan-to-scan registration for which the authors proposed a novel ICP metric. This metric contains symmetric Kullback-Leibler divergence (KLD) to cope with the differences between the distribution shapes of the geometric approximation. An algorithm favoring high accuracy over real-time performance was recently proposed in [58] for offline data processing by tightly-coupling computationally-heavy processes of error-state Kalman filter (ESKF), graph optimization framework g^2o , and generalized ICP together with normal distributions transform (NDT) for accurate point cloud registration.

■ 2.1.2 Geometrical Degeneracy

Although the field of LiDAR-based estimation has made great leaps in the recent decade (see Section 2.1.1), the performance of the methods is reliable only in well-structured environments. As shown in the DARPA SubT [7a], the ongoing challenges emerge in environments containing little to no amounts of salient geometrical structures. The lack of structure challenges LiDAR-based methods, as these methods exploit the local and global geometry of the environment. These scenarios most notably emerge in geometrically symmetrical environments, such as subterranean tunnels [4c, 32], caves [3c], and urban structures [8a, 31], as well as in historical monuments symmetrical along the vertical axis [1c, 2a, 10a] (see Figure 2.2). Such settings where the LiDAR point cloud contains few to no exploitable features (for at least one DoF that it is being optimized for) are denoted as degenerate. We further distinguish weak and strong variants of such degeneracy, as shown in Figure 2.1.

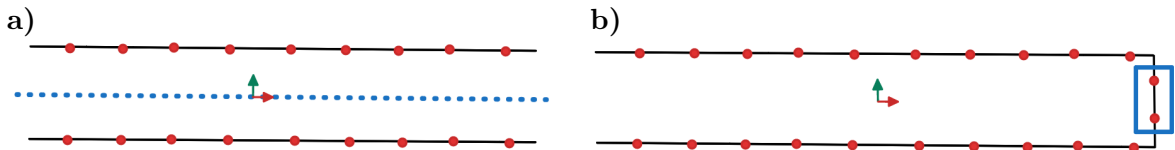


Figure 2.1: Simplified case of (a) **strong** and (b) **weak** geometrical degeneracy arising in robot-pose estimation problems. (a) Estimating motion along the blue axis is impossible using only the LiDAR point cloud given in red. (b) Only a handful of structures can be exploited for one of the DoFs (here in the horizontal translation axis). It is imperative that the area highlighted in blue is not classified as noise or as an outlier and is exploited in the optimization.

Apart from the lack of geometrical structures in the environment, the degeneracy also emerges from the limited field-of-view (FoV) of the LiDAR sensors³, especially when employing spinning-LiDAR technology. The narrow FoV reduces the perceptual observability of the

³30° vertical FoV for Velodyne Puck (velodynelidar.com/products/puck, accessed on March 27, 2024) and 22.5° for Ouster OS2 Rev7 (ouster.com/products/scanning-lidar/os2-sensor, accessed on March 27, 2024).

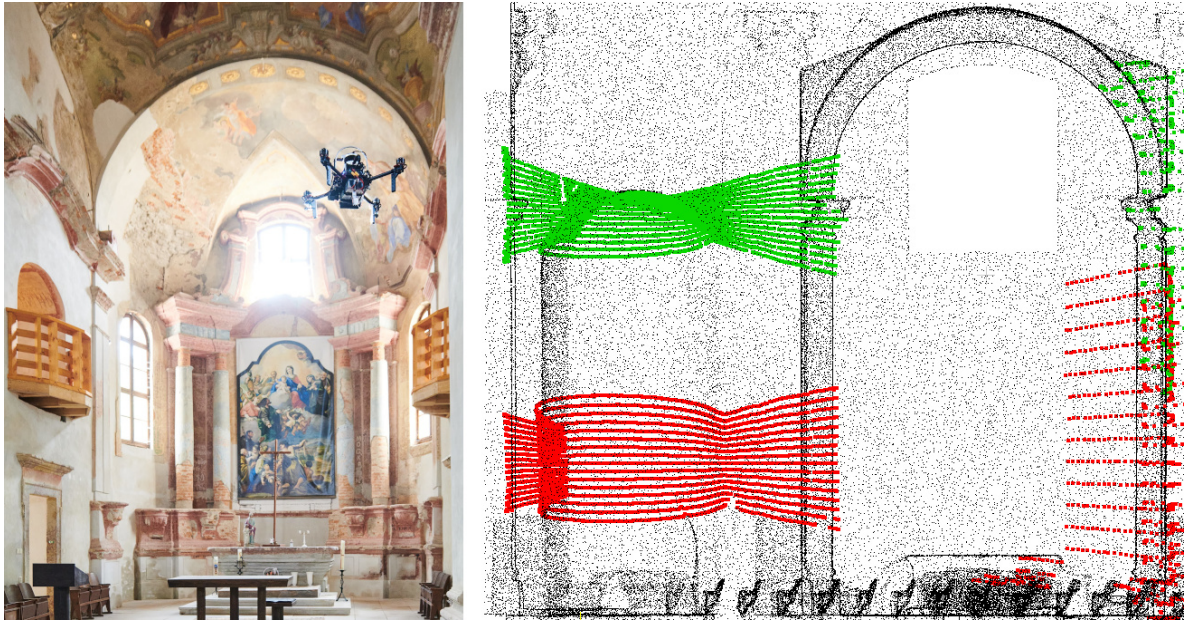


Figure 2.2: Geometrical degeneracy emerging in large historical monuments. By observing the left arc, the data in green are well-conditioned in all 6 DoF. However, the data in red contain only limited information (in the lower right part) that is exploitable for constraining the vertical axis. This problem, classified as weak degeneracy, is challenged in the core publications [1c, 6c].

geometric structures, even when these structures are present in the environment. The lack of observability then leads to drift in the ego-motion estimation. This is related to ill-conditioned directions in the context of solving the optimization problem, even when the environment is not inherently degenerate. This bottleneck has been tackled by the LiDAR manufacturers, which addressed the problem by developing wide-FoV hardware⁴. Although a wider FoV improves (but does not provide full) observability, it does not improve estimation in degenerate environments per se. In the related scientific literature, three main research branches tackle the problem of degeneracy — optimization for narrow LiDARs, the introduction of other sensory modalities, and the detection and mitigation of weak degeneracy.

Among the approaches optimizing for the narrow FoV of LiDARs is the LOAM-Livox [59], wherein the authors showed an improved performance if the points for feature extraction are selected in certain regions of the FoV with respect to the incident, intensity, and occlusion constraints. LoLa-SLAM [60] proposes to actuate the narrow-FoV LiDAR around two body axes to increase the space coverage while also compensating for the motion with a multi-threaded pipeline for matching scan slices. The challenges of spinning narrow LiDARs similarly also apply to solid-state LiDARs, for which the authors in [61] proposed one-stage odometry employing the scan-to-map methodology coupled with probabilistic mapping. The latter two approaches in tackling degeneracy through the introduction of other sensory modalities and weak-degeneracy detection are discussed in more detail in Sections 2.1.3–2.1.5.

⁴90° vertical FoV for Ouster OS0 Rev7 (ouster.com/products/scanning-lidar/os0-sensor, accessed on March 27, 2024).

■ 2.1.3 Multimodal SLAM

It is clear from Figure 2.1 that estimating the full robot state under strong degeneracy is impossible using just a single modality. Therefore, it has been proposed to introduce other sensor modalities complementing LiDARs in the estimation to tackle both the weak and strong degeneracy variants. These sources of modalities range from inertial measurements through cameras, wheel encoders, radars, and acoustic signals to radio-based ranging.

Among the related methods utilizing inertial measurements is LIO-mapping [62], which tightly couples LiDAR and inertial measurement unit (IMU) through joint-optimization and introduces a rotation-constrained mapping method to optimize the final poses and maps at the cost of computationally-heavy constraint construction and batch optimization. A similar tightly-coupled LI method LIOSAM [63] proposed smoothing over a factor graph to achieve highly accurate state estimation and mapping in real-time with efficient IMU pre-integration, bias correction, old scan marginalization, and keyframe selection. LINS [64] proposes extending LOAM with an iterated ESKF and an egocentric formulation in the odometry stage to reduce the drift, thereby improving the long-term state estimation. High accuracy is achieved by F-LOAM [65], employing the exact two-stage computation as in LOAM. In contrast to LOAM, the local feature maps in F-LOAM are proposed to be split for individual features (planes and edges) in order to include local smoothness in a proposed iterative optimization. A solution to overcome the geometrical-features degeneracy is applied in GR-LOAM [66]. Therein, a wheel encoder and an IMU of a ground robot were used to correct LiDAR-estimated ego-motion that tends to fail as the optimization problem is ill-conditioned from measurement aliasing. The fusion of features extracted from multiple non-overlapping LiDARs has been proposed to mitigate the geometrical degeneracy in the perception of LI-SLAM in MILIOM [67] and in LOCUS [68]. Both works employ similar techniques of graph-based optimization applied after time synchronization and the merging of data arriving from different sources.

Fusing LiDAR data with inertial information improves performance; however, it cannot be relied on long term if the LiDAR perception is degraded. This is because the drift in inertial data will inevitably accumulate from the integration of stochastic noise and the inability to correct biases with backpropagation of the state estimate that is obtained by a reliable source. To cope with this issue, the related literature increases the sensory modality with systems capable of state estimation independent of LiDAR information. Although these systems also suffer from perceptual degradation, the complementary information may improve the robustness of the entire estimation pipeline, as differing modalities suffer from distinct cases of degradation. Apart from inertial measurements, visual information in LiDAR-visual-inertial (LVI) methods is the most common modality. State-of-the-art literature commonly runs an instance of LiDAR-odometry and visual-odometry fusing the odometries together with inertial measurements to produce a state estimate. In contrast, high-level global optimization utilizes the odometries' information to compensate for their drift. Examples of such an approach include MIMOSA [69] and the method proposed in [70, 71], which applies a sequential multilayer coarse-to-fine processing with IMU-based motion prediction, a coupled visual-inertial (VI) method for motion estimation, and a scan matching for refinement and map registration. The sensor degradation (dark, texture-less for vision; geometrically structureless for LiDAR) is proposed to be bypassed with an arbiter that reconfigures the fusion pipeline with respect to a failure mode detection. The LVI methods of [70, 71] were recently extended with smoothing and mapping over an on-the-run built factor graph in LVI-SAM [72]. In R²LIVE [73], the authors proposed to lower the computational load of LVI methods by fusing fast LiDAR-odometry with efficient vision-based global optimization in the iterated

ESKF. Authors of LIC-Fusion [74] and its newer version LIC-Fusion 2.0 [75] had introduced plane-feature tracking in sliding-windows fashion, relying on a proposed online calibration of plane extraction and outlier rejection within data association. Reducing the computational load of LVI methods was proposed in [76], where the authors leverage GPU parallelization to match sequential data frames to the entire map using a dense factor graph (instead of a sparse pose graph) and utilize the generalized ICP metric efficient for its parallelization.

Accurate and computationally-heavy VIRAL-SLAM [77] performs BA over a local and a global factor graph with loop constraints, while utilizing LVI sensors and radio-based ultra-wideband (UWB) ranging. UWB ranging is also utilized in LIRO [78], where the authors leverage ranging to fixed landmarks in the environment during optimization. Lastly, deep learning methodology has been applied in LIO-CSI [79] to supply LIOSAM [63] with semantic information generated with a pre-trained neural network, and in SLOAM [80] to classify and parametrize trees in a forest used as semantic information in pose optimization.

Contributions: When designed well, multimodal approaches complement the advantages and disadvantages of the used modalities, thereby increasing robustness to the degradation of individual modalities. Nevertheless, such methodology yields three significant disadvantages. First, the methods run with increased computational costs emerging from processing other modalities, particularly in the case of visual sensors. Second, the detection of degeneracies needs to be implemented and tuned for each given modality, as a degenerate modality counteracts the overall estimation and negatively influences the final estimate. Lastly, the system is unable to accurately estimate when all of the modalities are simultaneously degenerate. For these reasons, our contributions advocate and propose two approaches for single-modal minimization of degeneracy effects. First, our proposed method for localizing a UAV in vertically-symmetrical historical monuments [1c] complements a low-FoV 3D LiDAR by improving observability along the degenerated axis with cheap and lightweight point-distance LiDAR sensors. Second, our redundancy-minimizing point cloud sampling [6c] proposes to maximize the quality of the extracted information from the LiDAR modality [6c] in order to lower the effects of weak degeneracy on the pose estimation performance (see Section 2.1.5). Moreover, in [1c] and our other works [2c–4a, 7a–12a], the proposed methodologies fuse the outputs of LiDAR-based methods with the inertial measurements to support the estimation pipeline running on-board a UAV. The fusion includes smoothing over a short history of measurements to cope with time delays in the pipeline using [81]. Although inertial measurements are noisy and subject to dynamic biases, they supply unstable long-term but high-rate information in-between frames of low-frequency LiDAR measurements (typically at 10–20 Hz). Therefore, the presence of inertial modality in these works is not a way to handle degeneracy, but to support the stability and control of a dynamically unstable multi-rotor UAV.

■ 2.1.4 Degeneracy-aware Sampling

The most common method to detect the degeneracy level of the LiDAR modality was introduced in [82]. In [82], the authors relate the geometrical degeneracy to information theory and define a degeneracy factor calculated from the eigenvalues of an information matrix of the optimization problem. This factor is used in quantifying the degeneracy in each optimized DoF. Moreover, this factor was also proposed in [82] to be used in analyzing the geometric structure of the problem constraints, thereby mitigating the convergence degeneracy by remapping the ill-constrained DoFs to another modality (typically inertial). Another factor is the localizability vector, which was proposed in [83] to evaluate the quality of the optimization problem with respect to the current measurements of a LiDAR and UWB ranging.

The method [83] utilizes the localizability vector in multi-modal sensor fusion to minimize the estimation drift within tunnel-like environments.

The factor introduced in [82] is utilized throughout the related literature to detect the per-DoF degeneracy and is used in lowering the effects of the degeneracy. The main branch of approaches proposes to sample (i.e., select, filter) the residuals used in the optimization in order to balance the optimization problem. The greedy-based method [84] defines the problem as a combinatorial optimization under a cardinality constraint. The method in [84] preserves the spectral attributes of the information matrix while applying greedy approximations to achieve computations in real time. The method in [84] selects the residuals such that the log-determinant of the approximate Hessian of the optimization problem is maximized up to a pre-specified threshold. KFS-LIO [85] does so similarly but maximizes the inverse trace of the approximate Hessian while one-by-one deleting the residuals from the problem’s Jacobian. The selection of a constant-size residual subset is proposed in [86] by buffering random-selected residuals for which the map correspondence leads to maximum enhancement of the objective, the log-determinant of the information matrix. X-ICP [87] filters out residuals with non-parallel plane normals per each DoF. The simplified version Xs-ICP [87] does this similarly to X-ICP but reuses the residuals computed in the first iteration as a prior in subsequent iterations. In particular, [84, 85, 87] utilize the eigenspace of the information matrix to quantify the degeneracy in the optimization, as introduced in [82]. The advantage of informed sampling (being performed at the optimization level) is the possibility of formulating awareness of degeneracy in the optimization. However, residual-space sampling requires the correspondence pairings to be known and is sensitive to noise in these correspondences, as well as to variability in point density. It also comes at the cost of re-sampling in every iteration of an estimation pipeline, as shown in Figure 2.3.

Contributions: On this note, our contributions propose an uninformed input-space sampling [6c], where the sampling is performed just once and without knowledge about the correspondence pairings. Without the correspondences, such sampling cannot relate to the information about the optimization problem as it is non-causal. Nevertheless, we discuss in Section 2.1.5 that input-space sampling can lower the effects of geometrical degeneracy in real-time systems, even when the correspondence pairings are unknown.

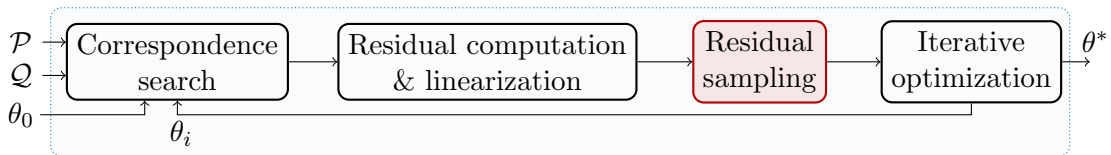


Figure 2.3: General architecture of an iterative estimation pipeline with sampling in the residual space. \mathcal{P} , \mathcal{Q} are two LiDAR point clouds and θ (typically $\theta \in SE(3)$) is the optimized variable (θ_0 is the initial estimate, θ_i is the iteration estimate, and θ^* is the output estimate). Sampling in the residual space requires the correspondences and their residuals and linearizations to be known. Computing all these iteratively is expensive, especially when the optimization gradients are low, resulting in high iteration counts (such as in point-to-point ICP, where high tens of iterations are common). In Section 2.1.5, the use of methods sampling prior to the iterative pipeline is advocated to remove this principal bottleneck.

■ 2.1.5 Point Cloud Sampling for Degeneracy Mitigation

Point cloud sampling preceding an iterative estimation pipeline is a common technique for reducing the complexity of the estimation problem. The typical architecture of such a pipeline is shown in Figure 2.4. Since the sampling reduces the cloud’s cardinality, the entire iterative pipeline is faster and more energy efficient. In theory, reducing the input point cloud cardinality lowers the number of required correspondence searches, which leads to fewer residuals and, thus, fewer linearizations and constraining factors in the optimization problem. Given that these expensive tasks are part of an iterative process, the overall computational gain can be significant (and even decisive) for mobile robots. However, this applies only when the point cloud sampling is computationally fast and preserves the quality of the points. While the former is subject to algorithm efficiency and available computational resources, the latter must preserve the overall information available in the point cloud. The latter, in particular, remains an ongoing research challenge.

Since modern 3D LiDAR sensors produce a huge amount of data (up to five million points per second), uninformed point cloud sampling is utilized in almost every state-of-the-art pipeline. The current approaches include point-density normalization [52] (i.e., voxelization), feature extraction [42, 88], normal-space sampling [89, 90], and learning-based inference [91]. The advantages and disadvantages of these methods are compared in detail in Section II. *Related Work* in [6c], which comprises the core of this thesis and is described in detail in Chapter 3. It has been shown that a point cloud sampling can improve the performance of real-time pipelines in well-conditioned settings [42, 52] if the sampling is fast and removes noise and outliers. Although sampling at the input of an estimation pipeline cannot relate to the optimization problem (i.e., degeneracy-aware factors [82, 83] cannot be used to quantify the contribution of an input point), it has been shown that uninformed methods may improve performance under weak geometrical degeneracy. This was first shown in [89, 90], where the authors utilize point normals to guide the sampling.

Contributions: Our novel method proposed in [6c] minimizes point redundancy within the point cloud and offers a novel alternative to typical point cloud sampling methods. By balancing the translational space, [6c] preserves the overall quality of points within the point cloud, even with no need for computing the point normals. In contrast to the state of the art, [6c] samples the least amount of points while yielding superior performance in well-conditioned settings, shows the highest drift reduction under weak degeneracy, speeds up the estimation pipeline the most, and preserves the highest amount of information. In contrast to other methods, [6c] adapts seamlessly to numerous diverse environments and is parametrizable by a single parameter, which relates to the estimation pipeline. It is also invariant on the environment, sensor, and the robot. Additionally, our contributions show that the sampling can be successfully applied in feature space (e.g., plane and line features [42]), complementing the verified and field-tested methods with novel research. Furthermore, the methodology proposed in [4c] includes a sampling of point clouds containing a high level of noise emerging from the clouds of dust whirling around the sensor. This novel method mitigates this degeneracy source by local sampling in the light-intensity field of modern 3D LiDARs. Our fundamental advances in point cloud sampling offer a quick, plug-in improvement to all existing methods employing rotating 3D LiDARs, even under degraded sensing.

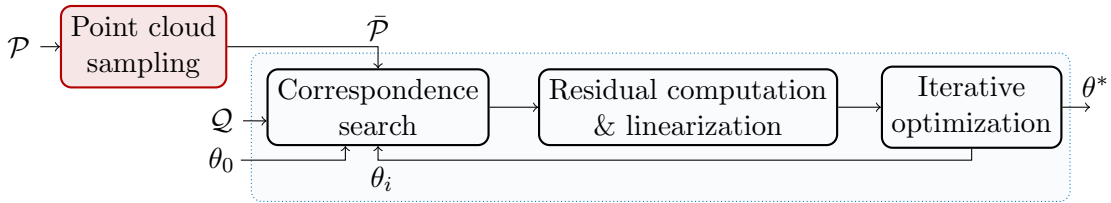


Figure 2.4: General architecture of an iterative estimation pipeline sampling in the input space. \mathcal{P} , \mathcal{Q} are two LiDAR point clouds (\mathcal{P} is the LiDAR point cloud and \mathcal{Q} is a target point cloud, such as a map), $\bar{\mathcal{P}} \subseteq \mathcal{P}$ is the sampled source point cloud, and θ (typically $\theta \in SE(3)$) is the optimized variable (θ_0 is the initial estimate, θ_i is the iteration estimate, and θ^* is the output estimate).

■ 2.2 Distributed Multi-Robot Sensing and Acting

Sharing an environment with a team of fast-flying UAVs requires synchronized coordination for precise and effective navigation. The research of this thesis focuses on two fundamental approaches to coordinating such aerial teams — decentralized swarming exploiting local perception and distributed coordination with wireless information sharing. The former tackles multi-robot coordination models utilizing solely local on-board perception of both homogeneous and heterogeneous robots as a substitute for centralized and computationally-heavy perceptual systems. The latter category then focuses on the effective and reliable use of multi-robot teams in tasks requiring multiple robots, such as S&R and aerial documentation.

■ 2.2.1 Perception-Aware Swarming

In recent years, several private companies (e.g., Intel[®], DAMODA) have aroused public interest in aerial teams by presenting visually attractive artistic light shows. However, the structural arrangement of robots in these artistic shows is centralized, with a ground station as the central element operating and supervising the show. Each robot follows a pre-computed trajectory in this configuration while relying on GNSS. This methodology originated from early literature within the research field of aerial swarming, such as [92]. Therein, the authors showed that a centralized computation unit could optimize the trajectories of 20 UAVs in order to reshape the swarm to a feasible (collision-free) formation given the states measured by a motion capture (MOCAP) system. Recent work [93] has also shown that this exact methodology can also be applied to a swarm where each agent employs a local state estimation (i.e., visual-inertial odometry (VIO)) if the central computation can transform all of the local state estimates into a global coordination system to align the pre-computed trajectories. Although the alignment in [93] was performed using a static MOCAP system, the state of the art in a collaborative SLAM [94, 95] shows that the swarm agents can co-localize without MOCAP systems by utilizing computations of map merging, BA, and location detection on a centralized server. However, the bottleneck of such architectures is the centralization element (i.e., ground station, server, communication), which serves as a single-point-of-failure.

A feasible methodology for removing this single-point-of-failure lies in the decentralization of the perception, computation, task reasoning, and sensory equipment. The research field of aerial coordination with purely decentralized architectures can be divided into methodologies of sharing or not sharing information among the team agents via a communication channel. Although communication brings advantages to team cooperation (e.g., sharing non-observable states, plans, and intentions), it reduces the system’s fault tolerance and limits the

scalability of dense teams where the robots need to sense and act in real time. To reduce the point of failure in communication, our focus lies on the methods acting solely upon local perceptual inputs. This motivation emerges from the field of perception-aware swarming, which follows the definition of an autonomous swarm as defined in [36] — a multi-agent team that is scalable for large groups, has high redundancy and fault tolerance, is usable in tasks unsolvable by a single robot, and has locally limited sensing.

Only a few works have developed and analyzed swarm systems following this exact definition [36]. In the past, the state of the art within this field was explored exclusively for 2D systems and, to this day, emphasizes self-organizing behaviors of simple units with minimal sensory capabilities, as introduced first in [96] for dimensionless particles and later summarized in [97]. One of these 2D self-organizing approaches is the Beeclust [98] algorithm, which utilizes probabilistic finite state machines and a primitive motion model to mimic the collective behavior of honeybees. The Beeclust algorithm has been verified in simulation as applicable to complex tasks not requiring information exchange among agents, such as in an underwater exploration of a seabed [99]. Another 2D approach [100] has shown that agents with limited sensing properties are capable of aggregation towards a common spatial goal while avoiding inter-agent collisions, similar to methods employing complete pose information. These communication-free decentralized algorithms, systematically described in [101, 102], require some form of mutual relative localization (even limited to binary detections) among the agents. Three-dimensional swarming in the real world has been achieved in [103–105]. In these works, the authors proposed decentralized flocking models utilizing local information and constraints on self-organization under the influence of communication, perturbations, and action delays in known confined environments. As a part of their analysis, the authors show that numerically optimizing the large set of parameters in their force-based flocking model may improve the stability of the swarm in real-world conditions, even for large flock sizes. However, the architecture in [103–105] lacks local perception and instead utilizes robot-to-robot communication to share the global states. Although recent improvement to multi-UAV navigation has been proposed in [106] by integrating localization and mapping onto small-factor robots, the methodology in [106] still communicates the planned trajectories.

The absence of accurate, robust, and reliable robot-to-robot perception techniques has hindered the research focus on aerial swarming in the real world. This has agitated the development of methods capable of localizing neighboring UAVs, which would enable perceptually-aware methods. Ranging from UWB radio signal has been proposed to obtain relative information among UAVs in [107]. From vision-based methods, multiple publications have employed active and passive infrared (IR) markers [108–111] since markerless perception in the visible spectrum is challenging. Machine learning methods utilized CNNs in detecting and localizing markerless agents in the visible spectrum [112–114]. Significant research on relative localization was also done at the MRS laboratory at CTU (affiliation of the author of this thesis), which began with an onboard vision-based methodology using passive circular markers [115] applied for stabilization of a formation of aerial agents in [116]. The preliminary research grew into the current state-of-the-art relative localization named UVDAR [37, 38], which proved to be a robust, reliable, and sufficiently accurate method enabling further state-of-the-art research in decentralized aerial swarming relying on direct localization of proximal agents.

Contributions: Our research in this area has utilized the UVDAR [37, 38] localization system in proposing swarming models capable of dynamic aggregation, navigation, and obstacle avoidance. In [5c], we have proposed a biologically-inspired swarming model designed for communication-free use on-board UAVs cooperating tightly in real-time within an environment filled with obstacles. The perception-aware model respects the constraints of a visual

system for relative localization (such as UVDAR [37, 38]) and proposes a set of local actions, which on the basis of local information yield self-organizing navigation of the aerial swarm. At the time of publishing, [5c] presented the first real-world UAV swarm capable of cohesive navigation without communication, all while respecting the definition of swarming from [36]. As part of our analyses, [5c] has also presented a study on the relation between the accuracy of position estimation and swarm stability. We showed that a zero-mean normal error in localization accuracy induces oscillations with an exponentially growing magnitude correlating with the deviation of the error. Although no information is shared among UAVs in [5c], the work utilized GNSS to self-localize each UAV in the world. Later contributions of the author and his colleagues then built upon [5c] by introducing novel swarming and flocking models to improve the obstacle [15a, 117] and predator [11a] avoidance, to enhance the cohesive [14a] and navigation [118] properties, and to apply proximal-control theory [16a] studied in the field of formation control. These contributions enhance the reliability of UAV-team deployments in safety-critical applications, such as the documentation of historic monuments [1c, 2c] discussed further in Section 2.2.2.

■ 2.2.2 Information-based Cooperation

In contrast to Section 2.2.1, the focus of this section lies in distributed multi-robot cooperation under continuous or intermittent communication. Given the spectra of the author’s contributions, emphasis is put on tasks of subterranean S&R and coordinated documentation of GNSS-restricted interiors by an autonomous team of ground and aerial robots.

The specified needs of S&R include a set of heterogeneous robots cooperating efficiently in exploring, assessing, and mapping devastated subterranean sites, such as collapsed mines and demolished buildings. The primary objective of such teams lies in assisting first responders by minimizing the risks of endangering their lives while providing a quick assessment and data intelligence from inside the surveyed site. The robot autonomy in this task must be capable of efficient (possibly optimal) operation in unknown, degraded, and dynamic environments. This includes autonomy in sensing, acting, decision making, coordination, and information retrieval, all under intermittent communication. The primary contributions in the related literature originate from the DARPA SubT, which was a robotic competition advancing the field robotics in S&R scenarios beyond the state of the art. The selected robotic systems developed during this competition are listed below in Section 2.3. A thorough summary of related work in this field is also part of our core work [4c], with emphasis on the particular subsystems of robotic autonomy.

On the other hand, inspecting interiors of a known environment by multi-UAV teams is another demanding task with its own challenges and caveats. This task requires accurate autonomous operation of a UAV in a known and safety-critical environment. Moreover, the nature of documentation techniques practiced within the field of cultural preservation requires autonomous teams of heterogeneous UAVs to operate synchronously, accurately, predictably, and safely in badly lit and confined areas high above the ground. In the related literature, a few approaches targeted single-robot autonomy utilizable for the specified domains, ranging from mobile ground robots [119] and solutions exploiting the known profiles of tunnels [32] and structured warehouses [120] to assistive systems [121]. However, before the contributions proposed in our publications, there was no robotic autonomy capable of autonomous multi-UAV operation in the interiors of industrial and historical settings for the purpose of their inspection and documentation. The related works in this field, including solutions proposed in the commercial sector, are summarized in our core work [2c].

Contributions: For the former task of S&R, our contributions include a top-performing stack for multi-UAV exploration of the targeted environments, described primarily in our core [3c, 4c] and other publications [8a]. In addition to the LiDAR-based perception described in Section 2.1.1, the contributions of this stack include novel methods for multi-UAV homing maximizing the limited flight time of robots, topometric map representations shareable among robots over a low-bandwidth intermittent network, full UAV-team autonomy in real-world and virtual environments, local [8a] and global path planning, and perception-aware exploration strategies. The contributions to the latter task of documenting historical monuments are part of our core works [1c, 2c] and preceding works [9a, 10a]. The works introduce a novel methodology for documentation tasks being performed by teams of autonomous UAVs in challenging safety-critical environments. In particular, the contributions include optimal coordination of heterogeneous robots (such as camera-equipped and light-equipped UAVs) in tasks requiring multiple robots in principle. The spectra of contributions of the author’s publications regarding this research problem are summarized in detail in Chapter 3 while further describing the core works of this thesis [1c, 2c].

■ 2.3 Theory to Practice: Validation of Fundamental Robotic Research

Field testing and experimentation are imperative to the field of mobile robotics. Real-world experimental analyses provide valuable information about the deployability of the robotic methodology and support the theoretical contributions proposed in a robotic work. This need for testing and experimentation is rooted in the fact that the applicability of the proposed methods may consequently prove to be limited when coupled with other systems since common theoretical assumptions often do not hold in the real world. This is especially true for real-time systems connected to UAV autonomy, where an overlooked detail can lead to catastrophic outcomes for the robot, human operators, and the environment. Although research in mobile robotics has commonly been limited to simulation analyses [121] and testing on offline datasets [63], there have been recent advances in real-world experimentation in state-of-the-art robotics. Field robotics has been advanced by individual works in the field [32, 33] but has also been greatly accelerated by the robotic competitions that aim to progress the boundaries of the entire robotic state of the art. The two most notable competitions are MBZIRC (held periodically every 3 years since 2017) and the DARPA SubT (2019–2021). MBZIRC focuses primarily on aerial robotics and pushes the boundaries of tasks unique to teams of UAVs, such as cooperative wall construction [122, 123], fire fighting [124, 125], capturing fast-flying objects [126, 127], and others. On the other hand, the DARPA SubT aimed to push the entire field of robotics by targeting the fully autonomous cooperation of heterogeneous robotic teams in completely unknown subterranean environments for the purposes of assisting first responders by providing situational awareness in disastrous scenarios. Targeting human-made tunnels, natural cave systems, and urban environments, the DARPA SubT led to the development of robotic stacks by the best robotic institutions. These stacks include [31] (winning solution by team Cerberus led by ETH Zürich), Nebula [128] (team CoSTAR led by Jet Propulsion Laboratory, NASA), [129] (team CSIRO Data61 led by the University of Queensland), and [4c, 12a] (team CTU-CRAS-NORLAB led by CTU, author’s team). The outcomes of the SubT in the perceptual domain are summarized in [7a], which concludes challenges persist in the state-of-the-art methodology, especially those related to perceptual degradations and solutions suited for resource-constrained robots. Both challenges are addressed in this thesis.

Contributions: Given the need for field testing and verifications, each of the papers published by the author (listed in Appendix A) has been actualized and validated in the real

world outside the laboratory. This unique approach shows the transferability of fundamental research into tasks applicable in unforgiving real-world conditions. The contributions to the real-world autonomy of UAV systems are highlighted for their imperative importance to field robotics, as they verify that the proposed theoretical ideas are robust, reliable, and accurate when interconnected with other onboard systems, each of which induces delays, errors, and assumptions. It also shows that the author's contributions are sufficiently robust and reliable for direct use in the real world, even in safety-critical applications of autonomous aerial inspection and documentation in historical structures [2c], as supported by utilization of the methodology in 18 historical monuments (including two sites on the UNESCO World Heritage list). This is supported by ongoing research in modular hardware solutions [13a, 17a] and software framework for reproducible research in the field of autonomous UAVs. The software framework of the MRS UAV System [39] (open-source at github.com/ctu-mrs/mrs_uav_system) is a shared project developed at the MRS laboratory at CTU that has been used worldwide for accelerating research and education as well as for minimizing the simulation-to-world bottleneck in aerial robotics. The author's contributions include additions to the open-source MRS UAV System, especially in 3D LiDAR-related methodologies, such as processing, sensor fusion, odometry, localization, SLAM, and mapping.

■ 3 LiDAR-based Perception in Demanding Real-World Conditions

- [1c] **P. Petracek**, V. Kratky, and M. Saska, “Dronument: System for Reliable Deployment of Micro Aerial Vehicles in Dark Areas of Large Historical Monuments,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2078–2085, 2020

The initial contributions to the established research problems have been introduced in [1c], a core work of this thesis. The work, published in the *IEEE Robotics and Automation Letters* and presented virtually at IEEE ICRA 2020, was motivated by the lack of robust and reliable localization for safety-critical purposes of documenting historical monuments with autonomous aerial vehicles. The work has proposed a then unique and novel method for equipping UAVs with lightweight 1D and 2D LiDARs for 6-DoF localization within a bounded 3D environment. The proposed method relied on onboard sensors and a 3D point cloud of the environment that was easily obtained from ground locations, with no need for a static localization infrastructure. The reliance on an apriori map of the environment allowed for the use of lightweight sensors while still keeping the system computationally undemanding enough to process data on-board UAVs with limited resources in real time. The allowance for lightweight sensors was essential at that time, as compact 3D LiDAR solutions without preposterous price tags were lacking presence on the market. Finances aside, lightweight sensors also allowed for the minimization of the robots. Although minimization would be optimally implemented with cameras, active LiDARs cope naturally with adverse and uncertain lighting conditions, require much less processing power as the data are sparser than camera images, and allow for easier, faster, and more precise geometrical mapping. The proposed localization method employed a multi-layered global-to-local localization with four asynchronous sensors — a 2D LiDAR, two 1D point rangefinders, and an IMU. The point rangefinders (oriented downwards and upwards on the UAV body) improve the sensory observability, and thus help in constraining the geometrical degeneracy along the vertical axis, which is common for tall historical monuments such as large cathedrals and churches. The performance of the proposed method was verified to reach the sufficient level of robustness and high precision required for accurate and repeatable documentation.

The localization method instantly found its applicability in [9a], where it served as an integral part of the novel task of reflectance transformation imaging (RTI) being performed by a team of autonomous UAVs. RTI is a photographic method for capturing an object’s surface shape and color, enabling interactive re-lighting of its virtual model from an arbitrary direction of illumination. This method is widely used in the field of restoration and cultural preservation, where the two works [1c, 9a] initiated their interdisciplinary overreach, later achieving successful scientific [2c, 10a], societal [130], and cultural [18a] achievements. In particular, [9a] has introduced an important contribution in transferring the previously resource-exhaustive RTI from the laboratory into the real world, where it was formerly inapplicable in large-scale settings and in areas not easily accessible by humans. Since RTI requires poses of all the UAVs to be known with respect to the documented object of interest, the localization method [1c] served a pivotal part of the aerial-RTI methodology. In [1c], the method also motivated the use of light-independent LiDAR technology since it benefits from minimum parasitic illumination and is optimally performed in total darkness, as later achieved in [2c].

Dronument: System for Reliable Deployment of Micro Aerial Vehicles in Dark Areas of Large Historical Monuments

Pavel Petráček^{ID}, Vít Krátky^{ID}, and Martin Saska^{ID}

Abstract—This letter presents a self-contained system for robust deployment of autonomous aerial vehicles in environments without access to global navigation systems and with limited lighting conditions. The proposed system, application-tailored for documentation in dark areas of large historical monuments, uses a unique and reliable aerial platform with a multi-modal lightweight sensory setup to acquire data in human-restricted areas with adverse lighting conditions, especially in areas that are high above the ground. The introduced localization method relies on an easy-to-obtain 3-D point cloud of a historical building, while it copes with a lack of visible light by fusing active laser-based sensors. The approach does not rely on any external localization, or on a preset motion-capture system. This enables fast deployment in the interiors of investigated structures while being computationally undemanding enough to process data online, onboard an MAV equipped with ordinary processing resources.

The reliability of the system is analyzed, is quantitatively evaluated on a set of aerial trajectories performed inside a real-world church, and is deployed onto the aerial platform in the position control feedback loop to demonstrate the reliability of the system in the safety-critical application of historical monuments documentation.

Index Terms—Aerial Systems: Applications, Aerial Systems: Perception and Autonomy, Localization

I. INTRODUCTION

IN recent years, massive advances have been made in the technology of aerial vehicles capable of vertical landing and takeoff, in terms of control, reliability, and autonomy. These multicopter vehicles, which we will refer to as Micro Aerial Vehicles (MAVs), have become extremely popular for their flexibility, diversity, and potential for applicability and amusement. The broad application spectrum of MAV systems ranges from 3-D mapping and deployment in search & rescue scenarios to wildlife & nature conservation.

This letter presents a unique self-localization system for a specialized use of MAV teams - autonomous documentation of historical monuments - derived from the interest expressed by end-users with expertise in restoration and conservation. The

Manuscript received: September 10, 2019; Revised December 11, 2019; Accepted January 9, 2020.

This paper was recommended for publication by Editor Jonathan Roberts upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by project no. DG18P02OVV069 in program NAKI II, by CTU grant no. SGS17/187/OHK3/3T/13, and by the Grant Agency of the Czech Republic under grant no. 17-16900Y.

The authors are with the Faculty of Electrical Engineering, Czech Technical University in Prague, 166 36 Prague 6, Czech Republic {pavel.petracek|vit.kratky|martin.saska}@fel.cvut.cz.
Digital Object Identifier (DOI): see top of this page.

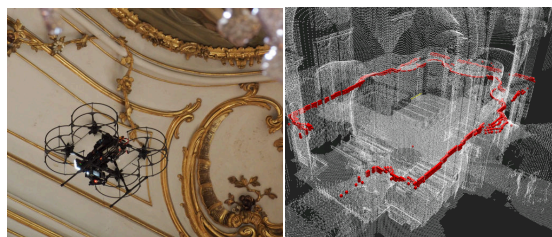


Fig. 1: Demonstration of the presented system during documentation of the parliamentary hall of the UNESCO castle in Kroměříž, Czech Republic, with an example of onboard sensory data registration into the object map

current procedure used during regular studies for restoration works requires a large scaffold to be constructed in order to monitor the condition of a building and its artifacts. An MAV platform can supply the same documentation and inspection techniques as those provided by the experts, but in locations unreachable by people except with the use of a large and expensive scaffolding installation, or in locations which had never been documented before during an initial survey. The MAV platform significantly speeds up the duration and significantly scales down the cost of the restoration works, and offers data acquisition from previously impossible angles and unreachable locations.

The proposed system is designed for deployment in historical monuments, such as ancient or modern, war-damaged, dilapidated or restored cathedrals, chapels, churches, mausoleums, castles, and temples with dimensions varying from small chapels up to large cathedrals. The deployment of robots in these operational environments is a challenging task due to the absence of a global navigation satellite system (GNSS), the adverse lighting conditions, and numerous other challenges summarized later on in Sec. III. An aerial system that handles all the challenges has to provide exceptional robustness, which we propose to achieve by introducing a precise model-based control approach, reliable real-time state estimation, a high level of sensor & actuator redundancy, and feasible mission planning & navigation.

In this letter, we also address in detail the problem of real-time state estimation acting as a state observer for MAV control and mission navigation modules in the tackled *GNSS-denied* environments of large historical buildings. The proposed system relies on a lightweight sensory setup composed of a 2-D laser-scanner and two point-distance rangefind-

ers, and on a map of a historical site pre-generated in the form of a 3-D point cloud provided by a terrestrial laser scanner (TLS). Our laser-inertial approach to indoor localization fuses an onboard IMU and a locally refined global state estimation, while it processes the data onboard an MAV and estimates the global state in real-time.

A. Related Work

Until now, documentation of interiors and exteriors of buildings and facilities has generally been performed manually by generating a 3-D site model using a TLS, together with scan registration post-processing [1] or photogrammetry [2] requiring geo-reference information. The emergence of automation procedures has speeded up the scanning processes. Examples are the *Zebedee* [3] and the *LIPS* [4] hand-held mobile 3-D laser mapping systems, and even ground robot mapping approaches [5].

Using aerial systems introduces the advantage of rapid documentation even in human-unreachable locations. These systems are being employed for documentation purposes [2], [6], but most of them are deployed manually outdoors and require GNSS to obtain geo-referenced data.

Although the lack of GNSS can be bypassed by exploiting a preset external localization system capable of a high accuracy localization, this approach is not scalable for documentation of large structures with limited access time. Other approaches for *GNSS-denied* localization include visual-inertial simultaneous localization and mapping (VI-SLAM), whose recent advances are well described in [7] and [8]. A mono- or stereo-camera SLAM is a thoroughly studied problem for an MAV, due to the lightweight of ubiquitous cameras. One example of a system that attempts to integrate online SLAM is the Open Vision Computer [9], which is an embedded off-the-shelf FPGA module that handles a visual SLAM independent of other onboard subsystems. The state-of-the-art vision-based ORB-SLAM2 [10] was tested in real conditions with lighting conditions similar to the desired environments. However, it was found to be ineffective, and it was disregarded for reasons described in Sec. III. The lighting issues motivated the development of the system presented here, which works under the specified conditions.

The authors in [11]–[13] have presented applications, which share a considerable number of common characteristics with indoor documentation of historical structures. A laser-inertial system for inspecting chimneys is presented in [11], while a laser-visual-inertial system for inspecting penstocks and tunnels is presented in [12]. In comparison to our application, environments tackled in [11] and [12] are well-structured and homogeneous for onboard sensors, which makes it possible to tune the system for these specific conditions. On the contrary, our task requires a much higher level of complexity. The authors in [13] focused on inventory applications in warehouses. Their laser-visual-inertial setup is suited for fast flights in complex dynamic environments in order to speed up a periodic inventory audit. However, our targeted scenario requires minimalist MAV dimensions, and slower and more accurate operation with respect to a variable onboard payload. In contrast to our lightweight sensory setup, the systems

in [11]–[13] employ a heavyweight 3-D lidar. Moreover, the systems in [12] and [13] fuse visual information from a set of onboard cameras, which is not suitable for the tackled environments with adverse lighting conditions.

Apart from that, only one work using MAV in the context of documentation of historical buildings has been found [14]. This work evaluates the performance of several SLAM and SFM methods during a 3-D model reconstruction of a single historical site. However, the authors of [14] perform only an offline evaluation of their methods on an outdoor aerial trajectory and do not deploy these methods in *GNSS-denied* environments nor in a position control loop of an MAV.

Documentation systems often extend their applications with a TLS to assist with the digital preservation of the scanned sites [15], which we likewise propose in our system architecture to boost the robustness of the system. In [16], map-based 3-D Monte Carlo localization (MCL) using an RGB-D camera provides a global state estimate. In contrast to this manuscript, our method utilizes a 2-D scanner instead of an RGB-D camera. This provides planar 360° information, making it independent from orientation. Our method goes further by refining the global estimate on a local map by a scan matching technique to yield faster and more accurate results. The authors of manuscripts [17], [18] fuse scan matching output, IMU, and a down-oriented rangefinder. Our proposed method extends the setup with global initialization and fusion of an up-oriented rangefinder. The importance of the up-oriented measurements rises significantly during flights over heterogeneous objects (church benches), where vertical estimate exploits the homogeneous nature of ceilings.

B. Contributions

This letter addresses problems of the deployment of aerial systems in the safety-critical application of historical monument documentation. The stability of the system originates from carrying out tests in real-world historical objects in the course of two years of a research and culture project in close cooperation with the National Heritage Institute of the Czech Republic. The insights into developments for real-world deployment presented here tackle the motivations and constraints of the highly challenging environments guided by end-users from outside the robotic community. The main contributions of this letter are:

- (i) It introduces a unique, highly reliable system for deployment in environments with low feature density and atrocious lighting conditions.
- (ii) It develops a unique hardware and software aerial platform designed in close consultation with restorers and conservationists, using experience from deployment of the system in numerous individual historical objects.
- (iii) It presents and shares the experience of what we believe to be the most comprehensive project in the field of autonomous documentation of historical monuments by an aerial system.
- (iv) It presents a robust light-independent localization system for interiors of historical buildings relying on 2-D lidar as its primary sensor.

The 3-D localization offers precise full 6 degrees-of-freedom estimation, providing fast and robust state estimation integrated into a feedback loop of an MAV position control. Based on a quantitative analysis evaluated on aerial ground-truth data in Sec. VI, our approach reaches persistent RMSE precision below 0.23 m. The drift-free system that is presented yields greater accuracy than map-based localization for autonomous cars [19] and comparable accuracy to a drift-prone method [20] employing a 3-D scanner on a ground vehicle.

II. MOTIVATION

The Dronument (Drone & Monument) project sets out to deploy MAVs for autonomous data acquisition in human-unreachable areas. The self-contained system presented here can be deployed in three modes (manual, semi-autonomous, and fully autonomous), as allowed by the heritage institute and/or the superintendent of the structure. These modes are specified as follows:

- (i) *manual*: a human operator controls all aspects of the flight using an operating transmitter, while the MAV is autonomously localized in the environment to associate gathered data with the 3-D map,
- (ii) *semi-autonomous*: a human operator commands the flight, while onboard systems control the sensory data acquisition and provide control feedback with respect to obstacles in a 3-D neighborhood, and
- (iii) *autonomous*: a human only specifies objects-of-interest (OoI) for documentation, and the system handles each stage of the entire mission - takeoff, stabilization & control, localization, navigation, trajectory optimization, data acquisition, and landing.

In addition to the deployment of a single MAV, the system is prepared for use in cooperative multi-MAV scenarios, as required for some documentation tasks. Typical non-invasive documentation consists of a multiple spectrum survey to obtain specific information valuable for various restoration purposes. For example, the use of different spectra contributes to more precise dating of paintings, as the glow of pigment combinations is unique to a certain period. Examples of single (S) and cooperative (C) tasks are:

- *Direct lighting (S)**: lighting of the scene from an onboard light with the illumination axis collinear with the optical axis of the camera.
- *Reflectance Transformation Imaging (C)*: a photographic technique for capturing the shape of a surface and the color of an object by combining photographs of the objects taken from a semi-static camera on an MAV under various illumination provided by a different MAV [21].
- *Three-point & strong-side lighting (C)**: filming techniques [22] in which 1-3 sources of light are used in different locations relative to the optical axis of the camera. In our previous work [23], a Model Predictive Control (MPC) approach is proposed for controlling a formation of MAVs with respect to the lighting techniques during an aerial deployment of cooperative teams in this task.
- *Radiography & UV screening (C)*: a method for viewing the internal structure of an object (e.g., a statue) by exposing it to X-ray or UV radiation (emission source

onboard the first MAV) captured behind or in front of the object by a detector (the second MAV).

- *3-D reconstruction (S/C)**: a method for aggregating the shape and the appearance of an object by combining laser- and/or vision-based information into a 3-D model.
- *Photogrammetry (S/C)**: a method for extracting measurements of real objects from photographs.

Examples of the tasks marked with (*) can be found within the additional multimedia materials available in [24].

III. EXPERIENCE GAINED

Over the last two years, more than 10 objects (mainly in Moravia, Czech Republic) were documented during the ongoing development phase of the presented system. Outputs of these documentation deployments supplied restorers and conservationists with valuable information in state assessment of multiple artifacts within the structures during the initial survey phase. Although some of the documented structures are shown in Figures 1, 2, and 5, the complete list, together with additional multimedia materials, can be found in [24]. During the experimental phase, many lessons from the robotic as well as the restoration point of view have been learned. The acquired experiences for objects of various sizes, shapes, and structures have influenced the system throughout the development and are herein shared.

The indoor surveys are conducted in the close vicinity of heavy-structure buildings. As a consequence, either GNSS is not available at all, or the system is not reliable enough, leading to *GNSS-denied* operations. In order to overcome the absence of GNSS, a local localization system must be used. This is true even when exteriors are being documented, with the intention to document facades in their close proximity in order to capture details of artifacts from various points of view (see exteriors documentation in [24]).

Insufficient *lighting conditions* in a surveyed object is an impediment for two main reasons. First, it degrades the performance of vision-based odometry and SLAM systems, which have been heavily researched over the last three decades. Second, it lowers the quality of the photographs taken in the visible spectrum, as they require decent lighting of the scene. These two issues motivated research in the Dronument project, leading to the development of a novel robust localization system (see Sec. V-B) purposely designed for autonomous flying in specific environments of this kind.

Experience has shown that vision-based localization is limited also by *feature extraction* shaped by two main characteristics. First, it is a common occurrence to fly along protracted segments of white wall lacking any visual features at all. Second, old religious buildings include extensive symmetric and repetitive visual patterns, such as grid flooring and artistic elements. A common example of such an artistic element are repetitive ledge supports shown on the right side of Fig. 2. Together with the lighting conditions, these considerations make most vision-based systems ineffective.

The use of a prior knowledge in the form of a *global map* obtained prior to the deployment of an MAV is extremely beneficial for three main reasons. First, it facilitates the robotic problem, supplies additional robustness to the system and

supports system reliability by serving as a baseline. Second, it yields an opportunity to associate captured onboard data with a 3-D map, which provides well-arranged data output for the end-user. Third, the visualization of the flight plan in a 3-D model is comprehensible for everyone - robotic experts, restorers, filmmakers - and it is necessary for confirmation purposes by an aviation authority, the heritage institute, and/or the administrator of the structure.

The proposed use of an MAV requires us to consider the *diversity of environments* in which it will operate. These environments contain distinct features - cluttered spaces, symmetric blueprints, balconies, stairs, glass windows, vault ceilings, and hanging strings. This forced the system design to include sensor and actuator redundancy, the use of a global map, and mechanical protection of the propellers. The test flights showed that the 2-D lidar that was used is ineffective for detecting thin obstacles, such as chandelier ropes and lighting cables. For this specific reason, a 3-D camera is employed to detect these obstacles in front of an MAV.

Even when MAVs are deployed in historical buildings, the presence of wind gusts ascribed to the stack effect (opened windows, doors) is non-negligible. To maximize the robustness of the system, particular emphasis must be laid on handling the *aerodynamic influence* of the MAV itself, and the wind gusts. First, the control subsystem must be resistant to these aerodynamic disturbances, in order to provide control stability (we rely on low-level stabilization designed in our team for flying in demanding desert conditions [25]). Second, perception modules must maintain their sensory properties when flying in low lighting conditions and in dust clouds, which originate when previously settled dust starts whirling.

An obvious constraint arises from the particular *historical value* of the surveyed objects and their invaluable character, which is the key reason for undertaking the documentation and restoration work. To avoid potential damage to fragile artifacts or to their surroundings at all costs, the MAV has to maximize the reliability and robustness attributes of the hardware and the software systems. The introduction of redundancy, system fault detectors, and safety procedures is a critical requirement for such a safety-challenging environment.

Last but not least, the system has to provide high *payload modularity*, in order to tackle all the documentation tasks in various environments. During documentation works, it may be necessary to document vertical walls and also the ceiling, even with multiple types of payload. For this reason, the hardware platform must be capable of rapidly changing the payload, its position, and the stabilization axes, in order to provide an effective solution to these end-user requirements.

IV. AERIAL PLATFORM

A specialized multi-MAV platform was developed to survey dark areas of historical monuments. This system is distributed to the primary MAV carrying the mission payload and a couple of lighter MAVs carrying additional mission equipment, such as lights. The primary MAV, shown in Fig. 2, is designed to minimize its dimensions, since the task assumes flights in narrow passages close to obstacles. Simultaneously, it is designed to maximize its payload weight capacities, since the payload is defined by the end-users, is interchangeable, and

often cannot be optimized for employment on aerial platforms. In its default configuration, the primary MAV carries an autopilot, an onboard computer, a down-oriented camera, two laser rangefinders, a 2-D laser scanner, a 3-D camera for obstacle detection, and the payload (an onboard light and a 2-axes stabilization hinge with a professional camera with its lens and a first-person view (FPV) system). The total weight is 5 kg, with a payload weight of 1.5 kg. To provide an extra level of safety, the platform is equipped with a mechanical propeller guard system to isolate the propellers from the external environment. The lighter MAVs carry a light with a digitally-controlled pitch angle (a degree-of-freedom required for formation trajectory optimization in [23]), light intensity, and color warmth.



Fig. 2: An application-tailored MAV carrying sensory and mission equipment during documentation of the Klein mausoleum in Sobotín, Czech Republic

V. SYSTEM ARCHITECTURE

The overall system architecture is composed of four main subsystems, which are hereafter described. The high-level pipeline of the system is outlined in Fig. 3.

A. Control Architecture

An MAV disturbance-resistant control pipeline was developed in the previous work of our group [25]–[28]. Beyond others, the MPC-based approach [26] was tested in the harsh environment of the desert in the United Arab Emirates during the MBZIRC 2017 competition, where it outperformed 147 registered teams [25], [28]. The system architecture presented in Fig. 3 is based on experience gained during this competition, which posed similar requirements of reliability, and resistance to wind disturbance and omnipresent dust. However, the task presented here is considerably different due to absence of GNSS and the density of the obstacles, and therefore goes beyond the work presented in [25] and [28].

B. Localization

For 6 degrees-of-freedom state estimation, we propose to rely on three laser-based sensors. First, a rigidly-mounted lightweight 2-D scanner produces data in the horizontal plane of the vehicle. Second, two point-distance laser sensors (rangefinders) measure the distance to the ground and ceiling objects. Together with an onboard IMU and a sparse 3-D map, the laser-inertial approach manages to estimate the global position and the attitude in light-independent conditions and without any heavyweight sensory equipment. The whole localization pipeline is summarized in Fig. 4, and will be described in detail in this section.

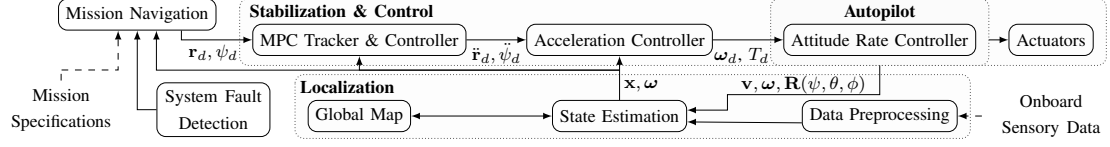


Fig. 3: High-level system pipeline of a single MAV. The stabilization & control pipeline [26] takes reference trajectory \mathbf{r}_d, ψ_d (points sequence of the desired 3-D position and yaw) as a setpoint for the MPC in the MPC tracker, which outputs a command $\dot{\mathbf{r}}_d, \dot{\psi}_d$ for the acceleration tracking SO(3) controller. The acceleration controller produces the desired angular rate ω_d and thrust reference T_d for the embedded attitude rate controller. The localization pipeline is described in detail in Sec. V-B.

1) *Global Map*: The localization system proposed in this work is designed to operate with a partially-known map registered from multiple 3-D intensity/color point clouds produced by a terrestrial laser scanner (TLS). The necessary granularity of the map (10 cm in our experiments) depends on the onboard computational resources and on the structural complexity of the documented building. Nonetheless, a modern TLS is capable of producing a scan with millimeter-level granularity, which makes the map subject to data reduction.

A raw map is processed by a set of filters (median filter, outlier rejection, uniform sampling) and is transformed to octree representation in order to employ optimized map operations, such as node traversal, integration of sensor measurements, data access, and tree node queries. During the preprocessing phase, artificial ground data is injected into the map to cope with missing data due to occlusions during scanning. Assuming that the ground is a cavity-free plane, the ground data is augmented by a set of points uniformly sampled from a plane. The parameters of this sampled plane are obtained from fitting it on a set of points withdrawn from the undermost parts of the available map using the RANSAC algorithm. Missing non-ground data is augmented in midair during a mission by the map refinement module.

A smaller structure requires < 5 scans, where a single full-dome scan (360° horizontal and 300° vertical field of view) takes approximately 3 min. During a field operation, there is enough time to produce a map of the object during the preparation of the equipment required for the mission. Example of a map is shown on the right side of Fig. 1 and in Fig. 7.

2) *State Estimation*: An MAV is assumed to have first-order dynamics for a short period of time during hovering and slow flights with negligible tilts (these flight characteristics are required in the confined areas in historical monuments for safety reasons). The linear stochastic discrete state-space model is used as

$$\mathbf{x}_{[k]} = \mathbf{A}_{[k]}\mathbf{x}_{[k-1]} + \mathbf{B}_{[k]}\mathbf{u}_{[k]} + \eta_{[k]}, \quad (1)$$

$$\mathbf{z}_{[k]} = \mathbf{H}_{[k]}\mathbf{x}_{[k]} + v_{[k]}. \quad (2)$$

The state $\mathbf{x}_{[k]}$, system input $\mathbf{u}_{[k]}$, measurements $\mathbf{z}_{[k]}$ and random noises of the system at time k are given as

$$\mathbf{x}_{[k]} = \left(\mathbf{p}_{[k]}^T, \boldsymbol{\Omega}_{[k]}^T \right)^T, \quad \mathbf{u}_{[k]} = \left(\mathbf{v}_{[k]}^T, \boldsymbol{\omega}_{[k]}^T \right)^T, \quad (3)$$

$$\mathbf{z}_{[k]} = \hat{\mathbf{x}}_{[k]}, \quad \eta_{[k]} \sim \mathcal{N}(0, \mathbf{Q}_{[k]}), \quad v_{[k]} \sim \mathcal{N}(0, \mathbf{R}_{[k]}), \quad (4)$$

where the state $\mathbf{x}_{[k]}$ is comprised from global position $\mathbf{p}_{[k]} = (x_{[k]}, y_{[k]}, z_{[k]})^T$ and attitude $\boldsymbol{\Omega}_{[k]} = (\psi_{[k]}, \theta_{[k]}, \phi_{[k]})^T$ consisting of the yaw, pitch, and roll angles; $\mathbf{v}_{[k]}$ is the linear

and $\boldsymbol{\omega}_{[k]}$ is the angular velocity of the IMU frame; $\hat{\mathbf{x}}_{[k]}$ is the measured global state; and $\mathbf{S}_{[k]}$ and $\mathbf{Q}_{[k]}$ are the covariance matrices of the process and the measurement noise at time k . The state-transition model $\mathbf{A}_{[k]}$, the control-input model $\mathbf{B}_{[k]}$, the observation model $\mathbf{H}_{[k]}$, and the covariance matrices $\mathbf{S}_{[k]}$ and $\mathbf{Q}_{[k]}$ are defined as

$$\mathbf{A}_{[k]} = [\mathbf{I}_{6 \times 6}], \quad \mathbf{Q}_{[k]} = \Delta t_{[k]} \begin{bmatrix} \boldsymbol{\Sigma}_{6 \times 6}^{mcl} & \mathbf{0}_{6 \times 6} \\ \mathbf{0}_{6 \times 6} & \boldsymbol{\Sigma}_{6 \times 6}^{icp} \end{bmatrix}, \quad (5)$$

$$\mathbf{B}_{[k]} = \Delta t_{[k]} \begin{bmatrix} \mathbf{R}(\psi_{[k]}, \theta_{[k]}, \phi_{[k]})_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{R}(\psi_{[k]}, \theta_{[k]}, \phi_{[k]})_{3 \times 3} \end{bmatrix},$$

$$\mathbf{S}_{[k]} = \Delta t_{[k]} \begin{bmatrix} \sigma_p^2 \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \sigma_\Omega^2 \mathbf{I}_{3 \times 3} \end{bmatrix}, \quad \mathbf{H}_{[k]} = [\mathbf{I}_{6 \times 6} \quad \mathbf{I}_{6 \times 6}]^T,$$

where $\mathbf{I}_{n \times n} \in \mathbb{R}^{n \times n}$ is an identity matrix and $\mathbf{0}_{n \times n} \in \mathbb{R}^{n \times n}$ is an empty matrix, $\boldsymbol{\Sigma}_{6 \times 6}^* \in \mathbb{R}^{6 \times 6}$ is the covariance matrix of the global and local state estimation, $\Delta t_{[k]} = t_{[k]} - t_{[k-1]}$ is the time elapsed since the last KF update, and $\mathbf{R}(\psi_{[k]}, \theta_{[k]}, \phi_{[k]}) \in \mathbb{R}^{3 \times 3}$ is the 3-D attitude. The presence of the rotation matrix in the control-input model copes with the differing global and IMU frames. The input of the system consists of inertial measurements coming at 100 Hz, and observations are produced by two estimation processes running in parallel, incoming at 5 and 20 Hz, which will be described below. The output of the KF correction step is equal to the output of the whole localization process.

3) *Monte Carlo Localization*: The configuration space of a robot inside an a-priori known map of a historical object is immense. This restricts the straight registration of sensory data to the extensive map due to the unknown initial conditions, which MCL provides in the form of a slow global state estimate. Concisely, MCL determines the posterior probability $p(\mathbf{x}|\mathbf{y}, \mathbf{u})$ of an unobservable state \mathbf{x} given sensor observations \mathbf{y} and control inputs \mathbf{u} by computing it on the state space subset in the form of hypotheses, yielding an approximation of the probability density function. The posterior probability can be obtained by employing the Bayes filter, which recursively computes the previous equation in the form of a belief $Bel(\mathbf{x})$ of the posterior probability as

$$Bel(\mathbf{x}) = \eta p(\mathbf{y}|\mathbf{x}) \int p(\mathbf{x}|\hat{\mathbf{x}}, \mathbf{u}) Bel(\hat{\mathbf{x}}) d\hat{\mathbf{x}}, \quad (6)$$

where η is a normalization constant. The derivation of the equation holds under the initial condition $p(\mathbf{x}_0) = p(\mathbf{x}_0|y_0, \mathbf{u}_0)$ and Markov independence assumptions.

Motion model: An odometry-based model for 2-D mobile robots employing the dead-reckoning principle is expanded to 3-D. In comparison with [29], our application requires slow

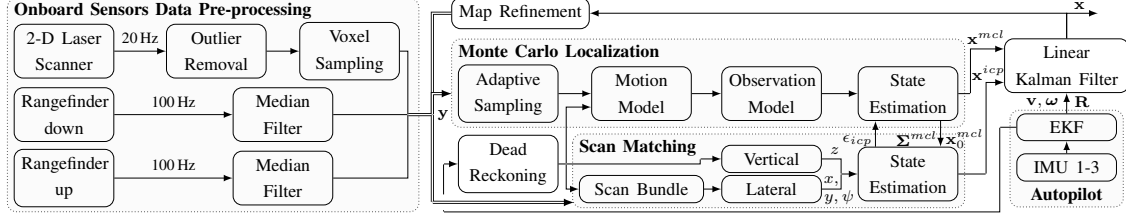


Fig. 4: Workflow diagram of the state estimation process. The laser-inertial pipeline fuses a global map, onboard data from multiple laser-based sensors (a 2-D horizontal scanner and an up- and down-oriented rangefinder), and 3 IMUs to produce state estimation \mathbf{x} . The IMUs are fused by an Extended Kalman Filter outputting linear \mathbf{v} and $\boldsymbol{\omega}$ angular velocity, and attitude $\mathbf{R}(\psi, \theta, \phi)$, where ψ, θ, ϕ are the Euler angles yaw, pitch, and roll. A decoupled local state refinement employing Iterative Closest Point is initialized after a global state estimate \mathbf{x}_0^{mcl} is provided by MCL. The validity of the state estimation processes is observed with respect to the MCL estimate covariance Σ^{mcl} and the absolute mean square error ϵ_{icp} of the scan matching.

movement of an airborne vehicle up to 0.5 m s^{-1} , making the variations in roll and pitch negligible and therefore allowing us to reduce the kinematic DoF to 4 (3-D position and heading).

Adaptive sampling: To improve performance, KLD-sampling [30] estimates the sufficient number of hypotheses M by bounding the error introduced by the sample-based representation of the MCL. The estimate is based on drawing from a discrete distribution with p different bins, and for

$$M \approx \frac{p-1}{2\epsilon} \left(1 + \frac{2}{9(p-1)} + \sqrt{\frac{2}{9(p-1)} z_{1-\delta}} \right)^3, \quad (7)$$

guarantees with probability $1 - \delta$ that the Kullback–Leibler distance between the maximum likelihood estimate (MLE) and the true distribution is less than ϵ , with $z_{1-\delta}$ being the upper $1 - \delta$ quantile of the normal $\mathcal{N}(0, 1)$ distribution.

To prevent convergence to an erroneous local minimum, a subset of hypotheses with the lowest weights is replaced in each resampling step with a dynamic-size set of new randomly generated hypotheses over the whole sampling space and a static-size set of new hypotheses matching the position of the latest state estimate with randomly sampled heading. The ratio of newly injected hypotheses is regulated by Augmented-MCL [31], which compares the short-term and long-term likelihood of observations as

$$M_{new} = M \max \left\{ 0, 1 - \frac{w_{fast}}{w_{slow}} \right\}, \quad (8)$$

where $w_{slow} = w_{slow} + \alpha_{slow}(\bar{w} - w_{slow})$ and $w_{fast} = w_{fast} + \alpha_{fast}(\bar{w} - w_{fast})$ for \bar{w} being the weighted average over the whole set of hypotheses, and $0 \leq \alpha_{slow} \ll \alpha_{fast}$ are the decay rates.

4) *Local Refinement:* To obtain precise and fast localization, local map registration is performed in a decoupled manner. The decoupling emerges from the sensory setup due to the vast difference between the data volume in the horizontal and in the vertical plane. In contrast to the vertical plane, where only two point-distance measurements are obtained, the horizontal sensor generally provides a greater number of samples (e.g., 16K samples per second for RPLIDAR A3), which needs to be reduced. The vast difference in the data volumes requires decoupling, otherwise the horizontal estimation would heavily overweigh the vertical estimation.

Lateral estimation employs a variant of the Iterative Closest Point (ICP) algorithm. Given a reference set of points \mathbf{P} and a target set of points \mathbf{Q} , the optimization process finds a transformation \mathbf{T} , which minimizes the weighted point-to-point error metric

$$E(\mathbf{T}) = \frac{1}{N} \sum_{i=1}^N (w_i \|\mathbf{T}\mathbf{p}_i - \mathbf{q}_i\|^2) \quad (9)$$

over the set of N correspondence pairs $(\mathbf{p}_i, \mathbf{q}_i, w_i)$, $\mathbf{p}_i \in \mathbf{P}$, $\mathbf{q}_i \in \mathbf{Q}$, $w_i \in \mathbb{R}$, $\forall i \in \langle 1, N \rangle$, $N \in \mathbb{Z}$. An initial solution to Eq. 9 is given by the dead-reckoning principle. Determining of the correspondence pairs involves closest distance pairing and a median filter, duplicate reference matches, and RANSAC-based pairs rejectors. Implementation of the ICP is based on the Point-Cloud library [32].

The reference scan \mathbf{P} is obtained onboard from a 2-D laser scanner, and its data are prepared according to Fig. 4. To provide improved robustness, a short history of the measurements is bundled together using short-time IMU-based dead-reckoning odometry, and is used as the reference scan \mathbf{P} for 3-D scan matching. An example of the scan bundle, registered into a map in the form of a 3-D point cloud, is displayed on the right side of Fig. 1. The target scan $\mathbf{Q} = \mathbf{Q}_{pla} \setminus \mathbf{Q}_{occ}$, $\mathbf{Q}_{pla} \in \mathbf{Q}_{map}$, $\mathbf{Q}_{occ} \in \mathbf{Q}_{pla}$, is derived from \mathbf{Q}_{map} and state estimate from the previous time step $\mathbf{x}_{[k-1]}$. The subset \mathbf{Q}_{pla} represents points of the map located in between two planes parallel to the x - y plane of the 2-D sensor frame at distance $\pm d_{pla}$ on the z axis of the same frame. The subset $\mathbf{Q}_{occ} \in \mathbf{Q}_{pla}$ represents all visually occluded points for which the linear path of a laser beam from a sensor position (rigidly defined by $\mathbf{x}_{[k-1]}$) to $\mathbf{q} \in \mathbf{Q}_{occ}$ is collision-free. A ray-casting algorithm, implemented over an octree representation of the map, is employed to determine the collision status.

During *vertical estimation*, a lateral estimate of the x , y axes, an attitude estimate, and the up- and down-oriented point-distance measurements are used to define a quadratic least squares problem

$$z^* = \arg \min_{z \in \mathbb{R}} (\alpha_{\uparrow}(\hat{\mathbf{p}}(z), y_{\uparrow}^r) \|y_{\uparrow}^m(\hat{\mathbf{p}}(z)) - y_{\uparrow}^r\|^2 + \alpha_{\downarrow}(\hat{\mathbf{p}}(z), y_{\downarrow}^r) \|y_{\downarrow}^m(\hat{\mathbf{p}}(z)) - y_{\downarrow}^r\|^2) \quad (10)$$

to find vertical z axis position z^* , where $\hat{\mathbf{p}}(z) = (x, y, z)^T$,

y_{\bullet}^r are real sensor data, and y_{\bullet}^m are map measurements found by map ray-casting. Bear in mind that the attitude and the rigid IMU-sensor transformations are neglected here to maintain simplicity. The validity coefficients α_{\bullet} are defined as

$$\alpha_{\bullet}(\hat{\mathbf{p}}(z), y_{\bullet}^r) = \begin{cases} 0, & \text{if } y_{\bullet}^m(\hat{\mathbf{p}}(z)) \text{ or } y_{\bullet}^r \text{ is invalid,} \\ 1, & \text{otherwise.} \end{cases} \quad (11)$$

Data invalidity emerges directly from invalid sensor measurements or from the absence of a map reference. In addition, the down-oriented sensor detects dynamic obstacles, such as people or map changes, which are observable from an identifiable discrepancy between real and map-based observations. These detections likewise classify the observations as invalid. In the case of $\alpha_{\uparrow} = \alpha_{\downarrow} = 0$, the z axis prediction at time k is given as

$$z_{[k]} = z_{[k-1]} + z_{[k]}^{imu} - z_{[k-1]}^{imu}, \quad (12)$$

where z^{imu} represents the integrated z axis position derived from the IMU-based dead-reckoning odometry.

C. Mission Navigation

To maximize robustness of the system, a visibility-constrained navigation is employed such that an MAV is allowed to maneuver only to obstacle-free areas visible from a front-facing depth camera. This approach supervises lidar-based perception by a redundant check for local obstacles in the camera field-of-view. An MPC-based control for navigation and trajectory optimization for MAV formations in the documentation task is introduced in our previous work [23].

D. System Fault Detection

In parallel to the mission controller, a tightly coupled fault detection system supervises all aspects of the mission. That includes supervision of the sensors and battery life status, of the state estimation covariance, or of the divergence from a preplanned trajectory. The whole system is implemented as a centralized high-level state machine capable of overriding the mission with an appropriate reaction to fault scenarios. Examples of these safety procedures are enforced controlled landing, trajectory execution termination, or manual take over of the control by a human operator. These safety responses can be likewise triggered by a mission operator, who is required to supervise the mission by an aviation authority.

VI. EXPERIMENTAL EVALUATION

To prove concept of the proposed method, the system was thoroughly verified in simulation (Gazebo 9 coupled with ROS Melodic), before it was deployed in position control feedback loop of an MAV. The main intention of the simulation was to estimate suitability of the developed system for deployment in safety-critical environments of historical buildings, to reduce probability of failures, and to obtain a qualitative analysis of the system behavior. Although the simulation results are omitted here due to lack of space, they can be found in [24].

A. Localization Precision Analysis

This section presents quantitative results of the localization system evaluated inside real church of St. Mary Magdalene in Chlumín, using a prototype MAV with the same sensory setup as is carried by the presented project platform. To obtain ground truth data, two Leica multi-stations were employed to track movement of the MAV equipped with the Leica GRZ101 360° Mini Prism reflector, as shown in 5a, which the stations are able to lock and track throughout 3-D space. Due to the lightweight and miniature dimensions of the particular reflector, the stations were capable to provide only the 3-D position of the reflector relative to a coordinate system of the stations at frequency of 5 Hz. The reference attitude was determined offline by ICP algorithm with parameters set to maximize accuracy. During short occlusions between a station and the target, a predicted trajectory of the target is followed in order to focus back once the occlusions disappear. Hence, the data further used as a ground truth reference contain short time period outages as the stations initialized re-locking procedure.

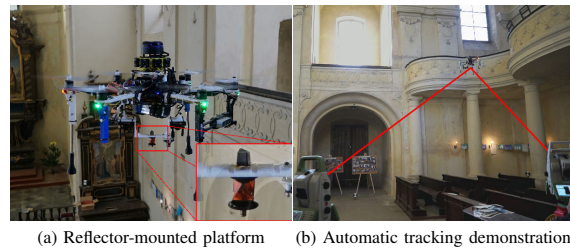
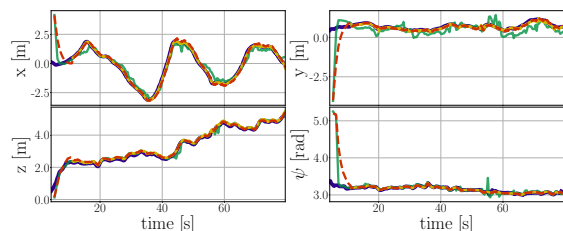


Fig. 5: An MAV platform equipped with onboard sensors and a reflector tracked by two Leica multi-stations measuring its 3-D position with precision of 1.5 mm at 5 Hz

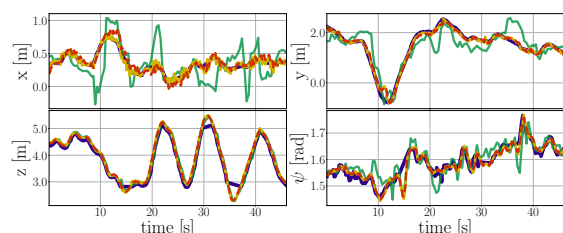
From multiple experimental flights tracked by an outer reference system, three particular trajectories are presented for which the quantitative results are given in Table I. Besides the table, outputs of the distinctive state estimation processes are outlined in Fig. 6 for the first two trials. The analysis of the localization system in real-world conditions exhibits estimation accuracy with translational RMSE less than 0.23 m during each experiment. The experiments also demonstrate minimal time delay, smoothness, and robustness of the state estimation. These attributes are important for reliable deployment as their absence could lead to destabilization of the MAV control. The proposed localization system proves to be a reliable and robust source with sufficient precision of the position estimate.

Trajectory	Trial 1	Trial 2	Trial 3
length [m]	24.055	45.812	21.163
avg linear velocity [m s^{-1}]	0.361	0.485	0.505
max linear velocity [m s^{-1}]	1.586	2.294	1.734
RMSE translation [m]	0.179	0.140	0.230
RMSE absolute orientation [$^{\circ}$]	2.381	2.460	2.747
max translation error [m]	0.385	0.522	0.594
max absolute orientation error [$^{\circ}$]	6.807	6.928	11.302

TABLE I: Quantitative results of the 3-D position and absolute yaw orientation ψ accuracy based on real data taken during real deployment in church of St. Mary Magdalene in Chlumín



(a) Trial 1: verification containing takeoff phase of the flight, where global estimation convergence and scan matching pipeline initialization is visible approx. at time 8 s and altitude of 2.300 m



(b) Trial 2: verification containing ground truth reference interruptions around 15 s, 30 s and 35 s due to visual occlusions between the multi-stations and the tracked target

Fig. 6: State variables x, y, z, ψ for ground truth (5 Hz), global (3 Hz) and local (15 Hz) localization, and fused state estimation (100 Hz) during real deployment in church of St. Mary Magdalene in Chlumín

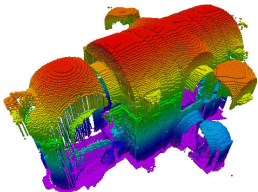


Fig. 7: Single-scan octree map of church of St. Mary Magdalene in Chlumín

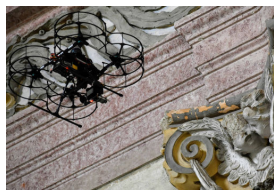


Fig. 8: Documentation in church of St. Anne and St. James in Stará Voda

VII. CONCLUSION

This letter presents the first comprehensive study on the use of autonomous MAV systems as an assistive technology for documentation of historical structures. The study shares the experience obtained during developing of the technology in close cooperation with team of restorers and conservationists, and discusses challenges of a robotic deployment. The proposed approach is validated and tuned on a set of identified tasks through extensive experimental flights aimed at collecting of exploitable data from the end-users.

To provide state estimate in *GNSS-denied* environments, an application-tailored localization system is presented, which was identified as the most important and challenging task in this application. This system provides local 3-D position and attitude without access to GNSS, and with the use of laser-inertial sensory setup copes with bad lighting conditions. That makes it feasible for deployment in indoor areas high above the ground, which are characteristic for historical monuments. The presented analysis of the localization system proves it to be a reliable and robust source of information with sufficient precision, which enabled its deployment into the feedback loop of the position control system.

REFERENCES

- [1] B. Kwoczynska *et al.*, "The Use of Terrestrial Laser Scanning in Surveying Historic Buildings," in *BGC Geomatics*, 2016.
- [2] D. Dominici, M. Alicandro, E. Rosciano, and V. Massimi, "Multiscale documentation and monitoring of L'Aquila historical centre using UAV photogrammetry," *ISPRS*, vol. XLII-5/W1, pp. 365–371, 2017.
- [3] R. Zlot *et al.*, "Efficiently capturing large, complex cultural heritage sites with a handheld mobile 3D laser mapping system," *JCH*, vol. 15, no. 6, pp. 670 – 678, 2014.
- [4] P. Geneva, K. Eickenhoff, Y. Yang, and G. Huang, "LIPS: LiDAR-Inertial 3D Plane SLAM," in *IEEE/RSJ IROS*, 2018.
- [5] D. Borrmann *et al.*, "Evaluation of Methods for Robotic Mapping of Cultural Heritage Sites," *IFAC CESCIT*, 2015.
- [6] D. Cowley *et al.*, "UAVs in Context: Archaeological Airborne Recording in a National Body of Survey and Record," *Drones*, vol. 2, p. 2, 2017.
- [7] M. Abouzahir *et al.*, "Embedding SLAM algorithms: Has it come of age?" *IEEE RAS*, vol. 100, pp. 14 – 26, 2018.
- [8] G. Huang, "Visual-Inertial Navigation: A Concise Review," in *IEEE ICRA*, 2019.
- [9] M. Quigley *et al.*, "The Open Vision Computer: An Integrated Sensing and Compute System for Mobile Robots," in *IEEE ICRA*, 2019.
- [10] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras," *IEEE Transactions on Robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.
- [11] M. Nieuwenhuisen *et al.*, "ChimneySpector: Autonomous MAV-based indoor chimney inspection employing 3D laser localization and textured surface reconstruction," in *IEEE ICUAS*, 2017.
- [12] T. Öztaşlan *et al.*, "Autonomous Navigation and Mapping for Inspection of Penstocks and Tunnels With MAVs," *IEEE RA-L*, vol. 2, no. 3, pp. 1740–1747, 2017.
- [13] M. Beul *et al.*, "Fast Autonomous Flight in Warehouses for Inventory Applications," *IEEE RA-L*, vol. 3, no. 4, pp. 3121–3128, 2018.
- [14] A. L. Majdik *et al.*, "Photogrammetric 3D reconstruction of the old slaughterhouse in Budapest," in *IWCIM*, 2016.
- [15] W. Sheng *et al.*, "Visual Point-Based Analysis of Laser-Scanned Historical Structures," in *IEEE ICC*, 2015.
- [16] F. J. Perez-Grau *et al.*, "An architecture for robust UAV navigation in GPS-denied areas," *JFR*, vol. 35, pp. 121–145, 2018.
- [17] G. A. Kumar *et al.*, "A LiDAR and IMU Integrated Indoor Navigation System for UAVs and Its Application in Real-Time Pipeline Classification," *Sensors*, vol. 17, no. 6, p. 1268, 2017.
- [18] F. Wang *et al.*, "An efficient UAV navigation solution for confined but partially known indoor environments," in *IEEE ICCA*, 2014.
- [19] Liang Wang and Yihuan Zhang and Jun Wang, "Map-Based Localization Method for Autonomous Vehicles Using 3D-LIDAR," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 276 – 281, 2017, 20th IFAC World Congress.
- [20] H. Ye, Y. Chen, and M. Liu, "Tightly Coupled 3D Lidar Inertial Odometry and Mapping," *CoRR*, vol. abs/1904.06993, 2019.
- [21] V. Kratky, P. Petracek, V. Spurny, and M. Saska, "Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles," *IEEE RA-L*, vol. 5, no. 2, pp. 2302–2309, 2020.
- [22] Y. Zhang *et al.*, "Lighting Design for Globally Illuminated Volume Rendering," *IEEE TVCG*, vol. 19, no. 12, pp. 2946–2955, 2013.
- [23] M. Saska, V. Krátký, V. Spurný, and T. Báča, "Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles using Model Predictive Control," in *IEEE ETEA*, 2017.
- [24] Multi-Robot Systems group, CTU FEE. (2019) Dronument (Dron & Monument). [Online]. Available: <http://mrs.felk.cvut.cz/dronument>
- [25] G. Loianno *et al.*, "Localization, Grasping, and Transportation of Magnetic Objects by a Team of MAVs in Challenging Desert-Like Environments," *IEEE RA-L*, vol. 3, no. 3, pp. 1576–1583, 2018.
- [26] T. Baca *et al.*, "Model Predictive Trajectory Tracking and Collision Avoidance for Reliable Outdoor Deployment of Unmanned Aerial Vehicles," in *IEEE/RSJ IROS*, 2018, pp. 1–8.
- [27] T. Baca, G. Loianno, and M. Saska, "Embedded Model Predictive Control of Unmanned Micro Aerial Vehicles," in *IEEE MMAP*, 2016.
- [28] T. Baca, P. Stepan, and M. Saska, "Autonomous Landing On A Moving Car With Unmanned Aerial Vehicle," in *IEEE EECMR*, 2017.
- [29] A. L. Ballardini *et al.*, "An effective 6DoF motion model for 3D-6DoF Monte Carlo Localization," in *4th Workshop on Planning, Perception and Navigation for Intelligent Vehicles, IROS*, 2012, pp. 121–126.
- [30] D. Fox, "KLD-sampling: Adaptive Particle Filters," in *NIPS*, 2001.
- [31] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. Cambridge, Mass.: MIT Press, 2005.
- [32] R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in *IEEE ICRA*, Shanghai, China, 2011.

- [6c] **P. Petracek**, K. Alexis, and M. Saska, *RMS: Redundancy-Minimizing Point Cloud Sampling for Real-Time Pose Estimation*, 2023. **Submitted to IEEE Robotics and Automation Letters (first decision *Revise and Resubmit* received on Jan 29, 2024)**

Another core publication [6c] of this thesis is currently under the review process in the IEEE Robotics and Automation Letters (first decision *Revise and Resubmit* received on January 29, 2024). The work [6c] tackles the problem of uninformed point cloud sampling (reduction) by defining and quantifying point redundancy in a 3D point cloud. Sampling point clouds is a necessary task that provides computational tractability in real-time estimation pipelines since the typical 3D LiDARs produce measurements (point clouds) with an abundant amount of data which have to be reduced. Such reduction has two main objectives — preserve the information extractable from the points and be computationally fast.

Our method [6c] brings novelty in defining redundancy in a point cloud and in the fast quantification and optimal minimization of such redundancy. This method is rooted in the derived theory, which shows that hyperplane surfaces generate redundant residuals in the translational space of the optimization. The theoretical and practical contributions of this method balance the translational input space of an estimation pipeline while preserving the information exploitable in the estimation task. Analyzing this method on data from UAVs, autonomous cars, and handheld devices has shown that in contrast to the state of the art, [6c] adapts seamlessly to divergent environments and outperforms the baseline methods in all relevant aspects. It is also parametrizable by a single parameter, which relates to the estimation pipeline (e.g., ICP, LOAM) and is invariant on the environment, sensor, and robot. It also samples the least amount of points, yet still yields superior performance in well-conditioned settings, shows the highest drift reduction under weak degeneracy (see Figure 3.1), and speeds up the estimation pipeline the most. Our contributions also show that the sampling can be successfully applied in feature space (e.g., plane and line features [42]), complementing the verified and field-tested methods with novel research. This fundamental advance in point cloud sampling offers a quick and plug-in improvement to **all** existing real-time estimation pipelines employing rotating LiDARs on-board any robot, including UAVs and self-driving cars. Since the method improves the speed and accuracy of real-time pipelines, it is of particular importance to our premise and the field of resource-constrained robotics. The method is available open-source at github.com/ctu-mrs/RMS together with a description video.

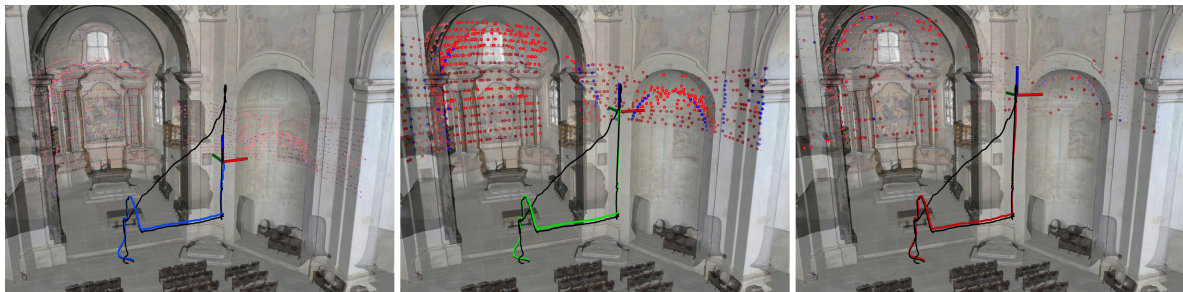


Figure 3.1: Performance of LOAM [42] in an environment illustrated in Figure 2.2 when preceded by three different point cloud sampling methods: voxelization (left), normal-space sampling [89] (middle), and proposed [6c] (right). While the proposed method reduces the estimation drift to a minimum, the estimation pipeline is also 122 % faster than with voxelization and 96 % faster than with normal-space sampling [89].

RMS: Redundancy-Minimizing Point Cloud Sampling for Real-Time Pose Estimation

Pavel Petracek*, Kostas Alexis*, and Martin Saska*

Abstract—The typical point cloud sampling methods used in state estimation for mobile robots preserve a high level of point redundancy. This redundancy unnecessarily slows down the estimation pipeline and may cause drift under real-time constraints. Such undue latency becomes a bottleneck for resource-constrained robots (especially UAVs), requiring minimal delay for agile and accurate operation. We propose a novel, deterministic, uninformed, and single-parameter point cloud sampling method named RMS that minimizes redundancy within a 3D point cloud. In contrast to the state of the art, RMS balances the translation-space observability by leveraging the fact that linear and planar surfaces inherently exhibit high redundancy propagated into iterative estimation pipelines. We define the concept of *gradient flow*, quantifying the local surface underlying a point. We also show that maximizing the entropy of the *gradient flow* minimizes point redundancy for robot ego-motion estimation. We integrate RMS into the *point*-based KISS-ICP and *feature*-based LOAM odometry pipelines and evaluate experimentally on KITTI, Hilti-Oxford, and custom datasets from multirotor UAVs. The experiments demonstrate that RMS outperforms state-of-the-art methods in speed, compression, and accuracy in well-conditioned as well as in geometrically-degenerated settings.

Index Terms—Localization, Range Sensing, Aerial Systems: Perception and Autonomy

MULTIMEDIA MATERIALS

The paper is supported by code and multimedia materials available at github.com/ctu-mrs/RMS.

I. INTRODUCTION

For the accurate and real-time ego-motion estimation of a resource-constrained robot, the amount of data provided in a 3D LiDAR point cloud is plentiful. To achieve convergence under real-time constraints (i.e., number of iterations, comp. time, convergence rate), the point clouds must be reduced. Apart from cardinality reduction, the objectives of such point cloud sampling are twofold — preserve the quality of the points and be computationally fast. While the latter is subject to algorithm efficiency and available computational resources, the former must preserve the overall information available in the point cloud. In the task of point cloud matching, the contribution (i.e., information) of a point has been shown to be quantifiable via its point-map correspondence and the shape of the loss function [1]–[4]. However, information-aware sampling of an input point cloud without the knowledge of these point-map correspondences (uninformed sampling) is non-causal and remains an ongoing challenge.

The work was supported by [to be added upon acceptance].

Authors are with the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University (CTU) in Prague, Czech Republic (*) and the Autonomous Robots Lab, Norwegian University of Science and Technology (NTNU), O. S. Bragstads Plass 2D, 7034, Trondheim, Norway (*). Corresponding author: pavel.petracek@fel.cvut.cz.

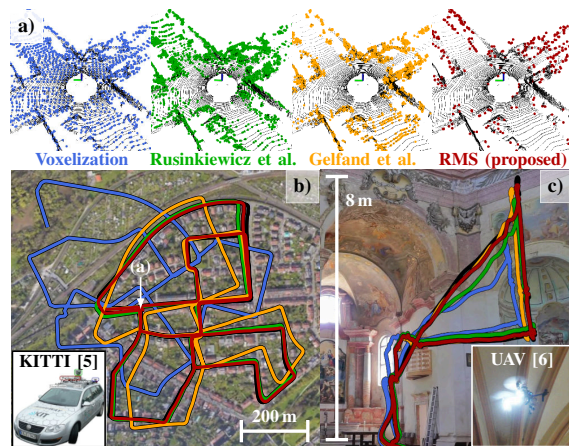


Fig. 1: A fast and noise-filtering 3D point cloud sampling can speed up real-time estimation pipelines. (a) An example of a single-frame sampling at the crossroad highlighted in (b) by each of the given methods (input point cloud in black). (b–c) Although sampling in the input space is uninformed about point-map correspondences, such sampling can improve performance if the sampling is fast and preserves the quality of the points (e.g., removes non-informative points). (b) Trajectory estimated by KISS-ICP [7] odometry on KITTI seq. #00 when preceded by one of the sampling methods (ground truth in black). Similarly, (c) shows trajectory estimated on-board a UAV using LOAM [8] odometry in a vertically self-symmetrical church in Stará Voda [6]. In LOAM, the plane and line features are sampled instead of the points.

The typical uninformed point cloud sampling methods include feature extraction [8], [9], point-density normalization [7], [10], normal-space sampling [11], [12], and learning-based inference [13]–[17]. With the individual advantages and disadvantages of these widely used methodologies, the overall challenges remain in their effectiveness, latency minimization, and environment adaptability. The experimental part of this paper shows that finding optimal parameters of such methods is often a balance between speed and accuracy. Our analyses also show that the optimal parameters are rarely adaptable to different LiDAR sensors, estimation pipelines, and environment types; and need to be exhaustively tuned for every instance.

Lastly, fast and noise-removing uninformed sampling has been shown to improve the performance of real-time pipelines in well-conditioned settings [7], [8]. However, it has also been shown that uninformed methods may improve performance in environments with a low amount of salient geometrical structures if these salient structures are part of the sampled data. We denote these settings, where the point cloud contains only a handful of exploitable structures, *weakly* degenerate. These settings most notably emerge in geometrically symmetrical environments, such as subterranean tunnels [18] and caves [19], and vertically-symmetrical historical monuments [6].

II. RELATED WORK

A ubiquitous point cloud sampling method is uniform sampling (voxelization), which discretizes space into fixed-sized cubes, each containing N (typically 1) points at maximum. Typical voxel-filter implementations employ an octree structure [20] or use a simple numerical discretization, such as that implemented in PCL [10]. The feature extraction methods consist of learning-based solutions (such as PointNet++ [9]) and hand-crafted feature (most commonly plane and line features defined in LOAM [8]) extractors. Although these methods perform reasonably well in geometrically rich settings when tuned properly, they are sensitive to parametrization. Moreover, learning-based methods lack sampling guarantees and require each environment to be part of the training data.

A deterministic sampling method [11] selects points such that their normals uniformly fill the normal-vector space. The covariance-based sampling (CovS) [12] iteratively selects the points, which maximize the expected normal-based contribution to the DoF least constrained in the eigenspace of the sampled set. Both methods [11], [12] have shown that point normals can be a helpful mechanism in guiding the sampling under *weak* geometrical degeneracy. However, obtaining the point normals cheaply, correctly, and reliably is challenging, especially given the projection nature of modern 3D LiDARs that generate data with uneven density and surface occlusions. PFilter [21] and ROI-cloud [22] are designed for use in a robot ego-motion estimation by employing previous LiDAR scans. PFilter [21] assigns each point a *persistence-index*, quantifying how persistent the point is over a short history of measurements. Static points, favorable in correspondence matching, tend to score higher in persistence. The ROI-cloud [22] divides space into cubes weighted by the amount of inlying geometrical features. [22] then propagates virtual particles representing past measurements and fuses them with the weighted cubes. Points are then sampled in areas where the weighted cubes align with the particles' distribution.

Among data-based sampling methods lies SampleNet [13], which learns task-specific sampling for object classification and geometry reconstruction. The method in [14] learns features and selects the points with the greatest contribution to the global max-pooling. DGCNN [15], FoldingNET [16], and KCNET [17] convert the point cloud into a graph and resample based on graph-based max-pooling, which takes the maximum features over the neighborhood of each vertex using a pre-built k-NN graph. The disadvantage of these methods is the absence of deterministic guarantees that the sampling will be invariant to the type of environment, and that it will maximize point relevancy in estimation.

Among the relevant redundancy-minimizing methods is [23]. Therein, the authors show that fewer correspondences are better in global registration, given that the correspondences are accurate. A map-compressing method [24] then applies concepts of feature similarity to select only one of the nearby features, marking the rest redundant and removing them. However, being formulated for expensive global registration and map compression, [23], [24] are inapplicable in front end of a real-time ego-motion estimation of a robot.

It has also been proposed that sampling is to be performed at the optimization level once the point-to-map correspondences are found. The greedy-based method [2] selects the optimization residuals such that the log-determinant of the approximate Hessian of the optimization problem is maximized. KFS-LIO [3] does so similarly, but maximizes the inverse trace of the Hessian. X-ICP [4] filters out residuals with non-parallel plane normals per each DoF, similarly to the normal-space equalization proposed in [11]. Simplified version Xs-ICP [4] does similarly to X-ICP but reuses the residuals computed in the first iteration as a prior in subsequent iterations. The advantage of sampling at the optimization level (informed) is the possibility to relate to the information theory, allowing to formulate awareness to degeneracy in the optimization. In particular, [2]–[4] utilize the eigenspace of the information matrix to quantify the degeneracy in the optimization, as introduced in [1]. However, residual-space sampling is sensitive to noise in correspondences and variability in point density and comes at a cost of re-sampling in every iteration of an estimation pipeline (see Fig. 2). Uninformed input-space sampling is computed only once per point cloud, but cannot directly relate to the degeneracy without the correspondence pairings.

The contributions of this paper include a novel out-of-the-loop 3D point cloud sampling named Redundancy-Minimizing Sampling (RMS). The method minimizes point redundancy within a point cloud by maximizing the entropy of the *gradient flow* in the sampled set. It builds upon the fact that hyperplane surfaces (i.e., linear and planar surfaces) contain a high level of redundancy propagated into the iterative estimation pipeline. Instead of classifying points into surface types, we propose a *gradient flow heuristic* (GFH) quantifying the potential of a point to lie on a hyperplane surface based on its local point distribution. The method is fast, uninformed, and deterministic and does not need point-normals to be known, is independent on the environment, is effectively parametrizable by a single parameter only, and is integrable into most state-of-the-art LiDAR-based odometries and SLAMs, both *dense* (using entire point clouds) and *feature-based*.

III. PROBLEM DEFINITION

The underlying problem of a six DoF robot ego-motion estimation from LiDAR data is scan matching. Scan matching can be formulated as finding the parameters $\theta^* \in SE(3)$, minimizing the squared sum of the residual functions $\mathbf{r} \in \mathbb{R}^3$ over two point sets $\mathcal{P} = \{\mathbf{p} \in \mathbb{R}^3\}$ and $\mathcal{Q} = \{\mathbf{q} \in \mathbb{R}^3\}$

$$\arg \min_{\theta \in SE(3)} g_{\theta}(\mathcal{P}, \mathcal{Q}) = \arg \min_{\theta \in SE(3)} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{C}_{\mathcal{Q}}^{\mathcal{P}}} \rho \left(\|\mathbf{r}(\theta, \mathbf{p}, \mathbf{q})\|_2^2 \right), \quad (1)$$

where $\mathcal{C}_{\mathcal{Q}}^{\mathcal{P}}$ represents the set of correspondence pairs from \mathcal{P} to \mathcal{Q} and ρ is a robust kernel with outlier rejection properties. Formulated as a pose estimation task, $\theta = \{\mathbf{t}, \mathbf{R}\}$ consists of a translation $\mathbf{t} \in \mathbb{R}^3$ and a rotation $\mathbf{R} \in SO(3)$ of the pose change from \mathcal{P} to \mathcal{Q} . Note that, \mathcal{P} and \mathcal{Q} can be entire LiDAR scans in *dense* or extracted features in *feature-based* formulations, and that the most prevalent \mathbf{r} functions in common iterative scan matchers are the *point-to-point*, *point-to-plane*, and *point-to-line* metrics, which are for a pair $(\mathbf{p}, \mathbf{q}) \in \mathcal{C}_{\mathcal{Q}}^{\mathcal{P}}$ given as

$$\mathbf{r}^{\bullet} = \theta \mathbf{p} - \mathbf{q}, \quad \mathbf{r}^{\square} = (\mathbf{n}^{\top} \mathbf{r}^{\bullet} + d) \mathbf{n}, \quad \mathbf{r}^{\mid} = \mathbf{r}^{\bullet} - ((\mathbf{r}^{\bullet})^{\top} \mathbf{v}) \mathbf{v}, \quad (2)$$

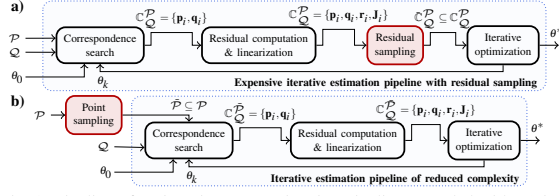


Fig. 2: Pipeline of an iterative pose estimation pipeline extended with (a) in-the-loop residual sampling and (b) a single-shot input data sampling. (a) The formulation (used in [2]–[4]) utilizes the full point cloud \mathcal{P} and introduces a significant overhead in each iteration. (b) The proposed architecture includes a single-shot out-of-the-loop sampling, which lowers the overall complexity by reducing the input size.

where (\mathbf{n}, d) is the parametrization of a plane that \mathbf{q} lies on (\mathbf{n} is a unit surface normal), \mathbf{v} is a unit direction of a line that \mathbf{q} lies on, and $\theta\mathbf{p} = \mathbf{R}\mathbf{p} + \mathbf{t}$.

In the related correspondence selection methods [2]–[4], the selection is formulated as finding a minimum-information correspondence subset that improves the performance of an iterative matching process in degenerate scenarios. Commonly, these works formulate the problem as a minimization task

$$\underset{\theta \in SE(3)}{\text{minimize}} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\mathcal{P}}} \rho \left(\|\mathbf{r}(\theta, \mathbf{p}, \mathbf{q})\|_2^2 \right), \quad (3)$$

$$\text{subject to } \mathbb{C}_Q^{\mathcal{P}} \subseteq \mathbb{C}_Q^{\mathcal{P}}, \mathbb{C}_Q^{\mathcal{P}} \neq \emptyset, \quad (4)$$

where $\mathbb{C}_Q^{\mathcal{P}}$ is a fixed-cardinality subset of correspondences selected from $\mathbb{C}_Q^{\mathcal{P}}$ with respect to the log determinant [2] or inverse trace [3] of the information matrix, and as a sum of constraints per optimization direction in the objective function [4]. Finding point-map correspondences $\mathbb{C}_Q^{\mathcal{P}}$ and then identifying the optimal subset $\mathbb{C}_Q^{\mathcal{P}}$ is expensive, especially when repeatedly computed within iterative algorithms.

Proposed formulation decreases the problem dimensionality by selecting points in the input scan \mathcal{P} before the iterative process of correspondence search, linearization, residual sampling, and optimization. We formulate the pose estimation as

$$\underset{\substack{\theta \in SE(3), \\ \bar{\mathcal{P}} \subseteq \mathcal{P}}}{\text{arg min}} g_\theta(\bar{\mathcal{P}}, \mathcal{Q}) = \underset{\substack{\theta \in SE(3), \\ \bar{\mathcal{P}} \subseteq \mathcal{P}}}{\text{arg min}} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\bar{\mathcal{P}}}} \rho \left(\|\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})\|_2^2 \right), \quad (5)$$

where $\mathbb{C}_Q^{\bar{\mathcal{P}}}$ is a set of correspondence pairs from $\bar{\mathcal{P}}$ to \mathcal{Q} and

$$\bar{\mathcal{P}} = \underset{\Omega \in \{\emptyset\} \cup \{\theta \subseteq \mathcal{P}, \theta \neq \emptyset\}}{\text{arg min}} |\Omega|, \quad (6)$$

$$\text{subject to } \underset{\theta \in SE(3)}{\text{arg min}} g_\theta(\mathcal{P}, \mathcal{Q}) = \underset{\theta \in SE(3), \bar{\mathcal{P}} \subseteq \mathcal{P}}{\text{arg min}} g_\theta(\bar{\mathcal{P}}, \mathcal{Q}). \quad (7)$$

In other words, we formulate the problem as finding a minimum-cardinality subset $\bar{\mathcal{P}} \subseteq \mathcal{P}$ over which the minimization problem converges to the same optimum as in the original formulation. Differences in iterative pipelines using formulations in Eq. (3) and Eq. (5) are shown in Fig. 2.

IV. INFORMATION REDUNDANCY MINIMIZATION

The problem formulated in Eq. (5)–(7) requires finding a minimal-cardinality non-empty subset of points $\bar{\mathcal{P}} \subseteq \mathcal{P}$ over which the estimation converges to the optimum without prior knowledge about the correspondences among point sets \mathcal{P} and \mathcal{Q} . This makes the formulation NP-hard and non-causal as the information about a point contribution to the optimization is unknown without its target correspondence. When a

correspondence $(\mathbf{p}_i, \mathbf{q}_i)$ is known, the related works [2]–[4] define its contribution in relation to the eigenspace of the information matrix $i\mathbf{J}_\theta^\top i\mathbf{J}_\theta$, where

$$i\mathbf{J}_\theta = \begin{bmatrix} \frac{\partial \mathbf{r}_\theta(\mathbf{p}_i, \mathbf{q}_i)}{\partial \mathbf{t}} & \frac{\partial \mathbf{r}_\theta(\mathbf{p}_i, \mathbf{q}_i)}{\partial \mathbf{R}} \end{bmatrix} \quad (8)$$

is the Jacobian of the residual function \mathbf{r}_θ (e.g., $\mathbf{r}_\theta^\square$ from Eq. (2) in [2]), or with relation to the approximate Hessian of the opt. problem [1] given as $({}^{\mathcal{P}}\mathbf{J}_\theta)^\top {}^{\mathcal{P}}\mathbf{J}_\theta$, where ${}^{\mathcal{P}}\mathbf{J}_\theta = \sum_{i=1}^{|\mathbb{C}_Q^{\mathcal{P}}|} i\mathbf{J}_\theta$. Although this makes the problem causal, finding the optimal minimum-cardinality subset is still NP-hard; and remains an open challenge.

To tackle this problem, we propose to approximate the solution to the problem formulated in Eq. (5)–(7) by defining, finding, and removing redundancy within a point set without knowledge about the correspondences. When applied to a typical iterative process of a robot’s ego-motion estimation, the proposed solution inherently removes noise and lowers the computational latency. When under real-time termination criteria (e.g., number of iterations, rate of change), the lowered cost improves the rate and accuracy of convergence.

A. Redundancy in a Point Set

Every perceived environment can be decomposed into a set of S atomic surfaces $\mathbb{S} = \bigcup_{s \in (1, S)} \mathbb{S}_s$ of arbitrary complexities, ranging from linear and planar to quadratic and other non-linear areas. In this work, the environment is assumed to be decomposable into linear and planar (hyperplane) surfaces. An input point set \mathcal{P} can then be understood as a discretization of the observed hyperplane surfaces $\mathcal{P} = \bigcup_{s \in (1, S)} \mathcal{P}_s$, where \mathcal{P}_s represents a set of points observed on the surface \mathbb{S}_s .

Definition 1. A single-surface point set \mathcal{P}_s contains information redundancy if removing one or multiple points from the set does not change its rate of information (average entropy) regarding the optimization problem.

Remark 1. In the optimization task defined in Eq. (1)–(5), the redundancy represents points that generate identical (parallel and of the same magnitude) residuals, whose removal does not alter the loss function, nor does it change the global optimum of the objective function.

Without the loss of generality, the robust kernel in Eq. (1)–(5) can be omitted for now, and a set-residual function (the objective function) can be defined as the sum of the point residuals

$$\mathbf{r}_\theta(\mathcal{P}) = \sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\mathcal{P}}} \|\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})\|_2^2 \quad (9)$$

to be minimized. Given the set of atomic surfaces \mathbb{S} and their corresponding point sets \mathcal{P}_s , Eq. (9) can be equivalently defined as a sum of surface-subset residuals

$$\mathbf{r}_\theta(\mathbb{S}) = \sum_{\mathcal{P}_s \in \mathbb{S}} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\mathcal{P}_s}} \|\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})\|_2^2. \quad (10)$$

Definition 2. Without altering the translational optimum, the objective function can be defined as a sum of set-residual rates

$$\bar{\mathbf{r}}_\theta(\mathbb{S}) = \sum_{\mathcal{P}_s \in \mathbb{S}} \bar{\mathbf{r}}_\theta(\mathcal{P}_s) = \sum_{\mathcal{P}_s \in \mathbb{S}} \frac{1}{|\mathbb{C}_Q^{\mathcal{P}_s}|} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathbb{C}_Q^{\mathcal{P}_s}} \|\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})\|_2^2. \quad (11)$$

Proof. The Jacobian of the obj. function defined in Eq. (9) is

$$\begin{aligned} \mathcal{P}\mathbf{J}_\theta &= \frac{\partial \left(\sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{C}_\mathcal{Q}^{\mathcal{P}}} \|\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})\|_2^2 \right)}{\partial \theta} \\ &= \sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{C}_\mathcal{Q}^{\mathcal{P}}} \frac{\partial \|\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})\|_2^2}{\partial \theta} = 2 \sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{C}_\mathcal{Q}^{\mathcal{P}}} \mathbf{r}_\theta(\mathbf{p}, \mathbf{q}) \end{aligned} \quad (12)$$

and the Jacobian of Eq. (10) is given as

$$\begin{aligned} \mathcal{S}\mathbf{J}_\theta &= \frac{\partial \left(\sum_{\mathcal{P}_s \in \mathcal{S}} \sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{C}_\mathcal{Q}^{\mathcal{P}_s}} \|\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})\|_2^2 \right)}{\partial \theta} \\ &= \sum_{\mathcal{P}_s \in \mathcal{S}} |\mathcal{P}_s| \frac{\partial \|\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})\|_2^2}{\partial \theta} = 2 \sum_{\mathcal{P}_s \in \mathcal{S}} |\mathcal{P}_s| \mathbf{r}_\theta(\mathbf{p}, \mathbf{q}), \end{aligned} \quad (13)$$

where \mathbf{r}_θ is a common residual for the redundant surface s . As each hyperplane surface s contains $|\mathcal{P}_s|$ identical residuals (see Def. 3), the simplification $\sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{C}_\mathcal{Q}^{\mathcal{P}_s}} \|\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})\|_2^2 = |\mathcal{P}_s| \cdot \|\mathbf{r}_\theta\|_2^2$ makes the Jacobians in Eq. (12) and Eq. (13) identical, assuming perfect point-to-surface associations. The Jacobian of Eq. (11) is derived similarly as in Eq. (12) and (13) as

$$\begin{aligned} \mathcal{S}\mathbf{J}_\theta &= \frac{\partial \left(\sum_{\mathcal{P}_s \in \mathcal{S}} \bar{\mathbf{r}}_\theta(\mathcal{P}_s) \right)}{\partial \theta} = \sum_{\mathcal{P}_s \in \mathcal{S}} \frac{1}{|\mathcal{P}_s|} \frac{\partial \left(\sum_{(\mathbf{p}, \mathbf{q}) \in \mathcal{C}_\mathcal{Q}^{\mathcal{P}_s}} \|\mathbf{r}_\theta(\mathbf{p}, \mathbf{q})\|_2^2 \right)}{\partial \theta} \\ &= \sum_{\mathcal{P}_s \in \mathcal{S}} \frac{1}{|\mathcal{P}_s|} \frac{\partial (|\mathcal{P}_s| \cdot \|\mathbf{r}_\theta\|_2^2)}{\partial \theta} = 2 \sum_{\mathcal{P}_s \in \mathcal{S}} \mathbf{r}_\theta. \end{aligned} \quad (14)$$

The Hessian matrices of all three formulations are given as

$$\mathcal{P}\mathbf{H}_\theta = 2 \sum_{i=1}^{|\mathcal{P}|} \mathcal{P}\mathbf{J}_\theta, \quad \mathcal{S}\mathbf{H}_\theta = 2 \sum_{s=1}^{|\mathcal{S}|} |\mathcal{P}_s| \mathcal{S}\mathbf{J}_\theta, \quad \mathcal{S}\bar{\mathbf{H}}_\theta = 2 \sum_{s=1}^{|\mathcal{S}|} \mathcal{S}\mathbf{J}_\theta, \quad (15)$$

where $\mathcal{P}\mathbf{J}_\theta$ is the Jacobian of \mathbf{r}_θ , as per Eq. (16)–(18) below.

The Jacobians of the residual functions (defined in Eq. (2)) are given analytically according to Eq. (8) as

$$\mathbf{J}_\theta^{\mathbf{I}} = [\mathbf{I}, -\mathbf{R}[\mathbf{p}]_\times], \quad (16)$$

$$\mathbf{J}_\theta^{\mathbf{n}} = [\mathbf{n}_i^\top \mathbf{n}_i, -\mathbf{n}_i^\top \mathbf{n}_i \mathbf{R}[\mathbf{p}]_\times], \quad (17)$$

$$\mathbf{J}_\theta^{\mathbf{v}} = [\mathbf{I} - \mathbf{v}_i \mathbf{v}_i^\top, -(\mathbf{I} - \mathbf{v}_i \mathbf{v}_i^\top) \mathbf{R}[\mathbf{p}]_\times], \quad (18)$$

given that $\mathbf{I} \in \mathbb{R}^{3 \times 3}$ and $\frac{\partial}{\partial \mathbf{R}}(\mathbf{R}\mathbf{p}_i) = -\mathbf{R}[\mathbf{p}_i]_\times$, where $[\mathbf{p}]_\times \in \mathbb{R}^{3 \times 3}$ is the skew-symmetric matrix of \mathbf{p} . It is clear that in the translational space (the $\partial/\partial \mathbf{t}$ part of the Jacobians), the change in residuals is, for the most common metrics, either constant or a function of the surface parameters. Since the translational change depends only on the surface s , selecting a single residual \mathbf{r}_s per surface preserves the basis of the translational eigenspace of both the Jacobian and Hessian matrices. Thus, the global optimum in the translational space of the objective function remains unchanged. ■

Remark 2. Although this reformulation does not alter the translational optimum, it reshapes the respective part of the objective func. g_θ to \bar{g}_θ without changing its monotonic intervals

$$\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^D, \quad g_\theta(\mathbf{x}) \odot g_\theta(\mathbf{y}) \Rightarrow \bar{g}_\theta(\mathbf{x}) \odot \bar{g}_\theta(\mathbf{y}), \quad (19)$$

where \odot is any linear inequality operator and D is the problem dimensionality.

Remark 3. When constrained to an ego-motion estimation task, we can assume the rotation changes to be small. Under this assumption, the first-order linearization of \mathbf{R} is given as

$\mathbf{R} \approx \mathbf{I} + [\mathbf{b}]_\times$, where $\mathbf{b} = [\alpha, \beta, \gamma]^\top$ is vector of the three rotational DoFs. Then, the rotational space in Eq. (16)–(18) reduces to a function of surface parameters and $[\mathbf{p}]_\times$, which denotes that sensitivity to rotations increases with point distance. This means that two points are also redundant in the rotational space if they belong to the same surface and have equal $[\mathbf{p}]_\times$.

Definition 3. Assuming zero noise, every hyperplane surface \mathcal{S}_s generates $|\mathcal{P}_s| - 1$ redundant residuals.

Proof. Given the hyperplane surfaces and their point-set observations \mathcal{P}_s , the set-residual rate $\bar{\mathbf{r}}_\theta(\mathcal{P}_s) = \bar{\mathbf{r}}_\theta(\Pi)$ applies for all $\Pi \in \{\pi \mid \pi \subset \mathcal{P}_s, \pi \neq \emptyset\}$. Eq. (14) then shows that reducing the cardinality of \mathcal{P}_s from $|\mathcal{P}_s|$ to 1 preserves the translational optimum, which implies that $|\mathcal{P}_s| - 1$ residuals are redundant. ■

Fig. 3 shows an idealized case demonstrating redundancy in surface-point sets, as defined in Def. 2 and 3.

Definition 4. Assuming the presence of noise, Def. 2 and 3 can be generalized to find the min-cardinality non-empty subset $\hat{\mathcal{P}}_s \subseteq \mathcal{P}_s$ whose set-residual rate matches the one of its superset

$$\hat{\mathcal{P}}_s = \arg \min_{\Omega \in \{\emptyset \mid \emptyset \subseteq \mathcal{P}_s, \emptyset \neq \emptyset\}} |\Omega| \quad (20)$$

$$\text{subject to } \bar{\mathbf{r}}_\theta(\Omega) = \bar{\mathbf{r}}_\theta(\mathcal{P}_s), \quad (21)$$

for each set of surface points \mathcal{P}_s . Given this formulation, each surface contains $|\mathcal{P}_s \setminus \hat{\mathcal{P}}_s|$ redundant residuals. Substituting Eq. (20) into Eq. (11) yields the obj. function in the form of

$$\mathbf{r}_\theta(\mathcal{S}) = \sum_{\mathcal{P}_s \in \mathcal{S}} \bar{\mathbf{r}}_\theta(\hat{\mathcal{P}}_s). \quad (22)$$

Remark 4. The reformulation is feasible since the Def. 2 maintains the convergence properties exploitable by the nonlinear solvers. The optimum consistency further satisfies Eq. (7).

In practice, the data are usually unstructured and are subjected to noise, making it expensive to segment the input point set \mathcal{P} into a set of surfaces, even trivially. Instead of finding and segmenting the underlying surfaces (as defined in Def. 3 and 4), we propose in Sec. IV-B a heuristic for the direct quantification of the redundancy without point-surface associations. In Sec. IV-C, we then propose a redundancy-minimizing algorithm robust towards noise, independent of correspondence matching, and invariant to small rotations.

B. Quantifying the Redundancy

As discussed at the beginning of Sec. IV, our objective is to find redundancy within a point set \mathcal{P} without knowing the correspondences $\mathcal{C}_\mathcal{Q}^{\mathcal{P}}$ beforehand. We tackle this by introducing a *gradient flow heuristic* quantifying the uniqueness of a point by local flow of a geometric gradient. The GFH maximizes the potential of points in bringing unique information to the optimization once their correspondences are found. Instead of expensive segmentation of the set \mathcal{P} into surface observations \mathcal{P}_s (as formulated in Def. 3 and 4), the GFH quantifies whether a point is locally a part of any hyperplane. Since generating multiple residuals on a single hyperplane is a source of the redundancy (as defined in Def. 1–4), this opens a way to the redundancy minimization discussed in Sec. IV-C.

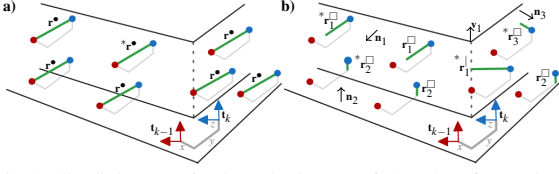


Fig. 3: Simplistic case of point redundancy (Def. 2 and 3) for a robot translating from the position t_{k-1} (● points) to t_k (● points). (a) The *point-to-point* metric generates identical residuals, which makes the residual rate constant for any positive number of residuals used, e.g., a single residual generates $\bar{\mathbf{r}}(\mathcal{P}) = \mathbf{r}^*$. (b) The *point-to-hyperplane* metrics generate identical residuals per surface (in this example, the surfaces comprise three planes and a single line). The minimized objective function remains constant if any positive number of points is sampled per each surface. In this case, using the minimum amount of samples yields the residual rate $\bar{\mathbf{r}}(\mathcal{P}) = \mathbf{r}^* + \sum_{i=1}^3 \mathbf{r}_i^*$. This example assumes perfect correspondences, which is unrealistic under noise and rotation. The point sampling method proposed in Sec. IV-B is designed to be robust to cases where this assumption is not met.

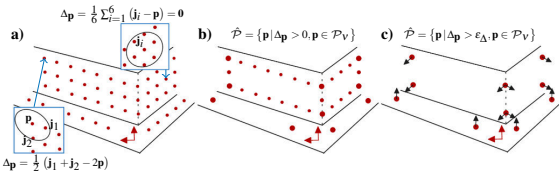


Fig. 4: Proposed *gradient flow heuristic* for quantifying redundancy in a point set. (a) GFH is computed for each point in a voxelized point set \mathcal{P}_v . (b) Points on the perceived borders have generally non-zero Δ_p , whereas (c) corner points yield the maximum Δ_p (herein thresholded by an abstract value ϵ_Δ). Keeping the $\max\text{-}\Delta_p$ subset (c) ensures that all directions remain constrained, as shown by the black axes representing which translational directions the points constraint.

The GFH emerges from Def. 2 and 3, which define that identical (in orientation and magnitude) residuals are redundant in structuring the objective function and that on a single hyperplane, the residuals are identical inherently. To quantify the uniqueness of points (and thus, the potential of future residuals), the GFH finds the neighbors of each point $\mathbf{p} \in \mathcal{P}$ within a spherical neighborhood with radius λ_p (m)

$$\mathcal{N}_p = \{\mathbf{j} \mid \|\mathbf{j} - \mathbf{p}\|_2 < \lambda_p, \mathbf{j} \neq \mathbf{p}, \mathbf{j} \in \mathcal{P}\}, \quad (23)$$

and defines the *gradient flow* (in meters) as

$$\Delta_p = \frac{1}{|\mathcal{N}_p|} \sum_{\mathbf{j} \in \mathcal{N}_p} \mathbf{j} - \mathbf{p}. \quad (24)$$

As demonstrated in Fig. 4, the GFH scores high for points lying on the borders of a surface and low for points lying inside the borders (inliers). Maximizing GFH thus leads to prioritizing the borders of surfaces rather than the surface inliers, which is important for two reasons. First, the borders in \mathcal{P} have the largest potential for correct correspondence matching with the borders of the corresponding physical surface. Second, the inliers can generate erroneous local minima and resist sliding along the directions of a hyperplane when using the *point-to-point* metric. As discussed in [25], the *point-to-hyperplane* metrics do not suffer from this deficiency, but it is still valuable to remove the redundancy to increase efficiency.

Every point-set matching algorithm more or less voxelizes the input set by a constant voxel size factor ν in order to reduce the cardinality of the input point set to $\mathcal{P}_v \subset \mathcal{P}$. We exploit this by setting $\lambda_p = 2\nu$ for unstructured point sets. In structured

point sets coming from a rotating 3D LiDAR (e.g., Ouster), the neighborhood radius instead respects the projective properties of these sensors as

$$\lambda_p = 2 \max \left[\nu, \|\mathbf{p}\|_2 \max \left(\sin \frac{2\pi}{C}, \sin \frac{\theta_v}{R-1} \right) \right], \quad (25)$$

where θ_v is vertical and 2π is a horizontal field of view of the sensor, which data are generated in a matrix form with R rows and C columns.

The neighborhood search of Eq (23) is the only expensive part of the proposed methodology. We construct a KD-tree from the voxelized point set \mathcal{P}_v to lower the cost. Using \mathcal{P}_v lowers the construction cost of the KD-tree and reduces the number of KD-tree queries to $|\mathcal{P}_v|$. With construction complexity $\mathcal{O}(n \log n)$ and worst-case complexity of n -query radius search being $\mathcal{O}(n^2 \log n)$ (where n is $|\mathcal{P}|$ in the full and $|\mathcal{P}_v|$ in the voxelized case), the overall complexity is reduced since $|\mathcal{P}_v| < |\mathcal{P}|$. We show in Sec. V that the overhead for computing the GFH for all the points lowers the complexity of the ego-motion estimation and accelerates the full pipeline.

C. Removing the Redundancy

Although a redundancy might be beneficial for reducing the effects of noise and outliers, it makes the iterative process of correspondence finding, residual generation, linearization, and optimization more expensive. Under the presence of termination criteria, the process may be undesirably hindered by accurate in-time convergence.

To find a solution to the NP-hard problem formulated in Eq. (20)-(22), we could propose to solve an optimization task

$$\hat{\mathcal{P}} = \underset{\Omega \in \{\Theta \mid \Theta \subseteq \mathcal{P}_v, \Theta \neq \emptyset\}}{\text{arg min}} \Gamma_\Delta(\Omega), \quad \text{subject to } |\Omega| = N_\Omega, \quad (26)$$

minimizing redundancy Γ_Δ in the *gradient flow* of the subset Ω under a constraint on fixed cardinality of the output set $N_\Omega \in (1, |\mathcal{P}_v|)$. Although the concept of a fixed-cardinality constraint is common within the related works [2], [3], the notion of redundancy allows for a more rigorous formulation. Since minimizing redundancy in data can be understood as maximizing the data entropy, we instead define a dual task

$$\hat{\mathcal{P}} = \underset{\Omega \in \{\Theta \mid \Theta \subseteq \mathcal{P}_v, \Theta \neq \emptyset\}}{\text{arg max}} H_\Delta(\Omega), \quad \text{subject to } \bar{H}_\Delta(\Omega) \leq \lambda_{\bar{H}}, \quad (27)$$

maximizing the entropy of information H_Δ in the *gradient flow* of the subset Ω under the termination criteria on the relative information rate \bar{H}_Δ (defined in Eq. (30)), given a maximum relative entropy rate $\lambda_{\bar{H}}$ (%). The termination criteria in Eq. (27) replaces the constraint on a fixed cardinality, which allows the selection to emergently adapt to the distribution of the points, making this method invariant to the type of environment. By thresholding the relative information rate via $\lambda_{\bar{H}}$, a certain level of redundancy is introduced into the system, possibly increasing robustness towards noise and outliers.

Let an entropy rate be an average entropy per point in set Ω

$$\bar{\mu}_\Delta(\Omega) = \frac{1}{|\Omega|} H_\Delta(\Omega), \quad (28)$$

where the entropy of the set is given as

$$H_\Delta(\Omega) = - \sum_{\mathbf{p} \in \Omega} p(\Delta_p) \log p(\Delta_p), \quad (29)$$

and p represents the probability of observing the GFH value $\Delta_{\mathbf{p}}$. The relative entropy rate (conditioned in Eq. (27)) is defined as the normalized entropy rate

$$\bar{H}_{\Delta}(\Omega) = \frac{1}{\bar{\mu}_{\Delta}^*(\Omega)} \bar{\mu}_{\Delta}(\Omega), \quad (30)$$

where

$$\bar{\mu}_{\Delta}^*(\Omega) = \max_{\Psi \subseteq \{\Theta \mid \Theta \subseteq \Omega, \Theta \neq \emptyset\}} \bar{\mu}_{\Delta}(\Psi) \quad (31)$$

represents the maximum entropy rate of all non-empty subsets $\Psi \subseteq \Omega$. Note that the entropy rate is an inverse function to redundancy within the data, which allows us to formulate the dual task in Eq. (27).

The probability function p is a function of the data \mathcal{P}_v , which are a function of the environment. To maintain invariance to the environment, p can not be modeled with a probability density function. Instead, we propose to use a frequentist's approach to approximate the probability function p . First, the GFH of each point in \mathcal{P}_v is converted to its normalized norm

$$\bar{\Delta}_{\mathcal{P}_v} = \left\{ \frac{\|\Delta_{\mathbf{p}}\|_2}{\max_{\mathbf{p} \in \mathcal{P}_v} \|\Delta_{\mathbf{p}}\|_2} \mid \mathbf{p} \in \mathcal{P}_v \right\}, \quad (32)$$

where $\max_{\mathbf{p} \in \mathcal{P}_v} \|\Delta_{\mathbf{p}}\|_2$ represents the maximum $\|\Delta_{\mathbf{p}}\|_2$ of any point $\mathbf{p} \in \mathcal{P}_v$. Second, a histogram \mathcal{H}_{Δ} with K bins bounded in interval $\langle 0, 1 \rangle$ is created out of the normalized GFH norms $\bar{\Delta}_{\mathcal{P}_v}$, where each bin $k \in \langle 1, K \rangle$ holds a point set ${}^k\mathcal{H}_{\Delta}$. The probability of a bin k is then approximated by $p_k = \frac{|{}^k\mathcal{H}_{\Delta}|}{|\mathcal{P}_v|}$.

In Alg. 1, we propose a point sampling routine following formulation in Eq. (27). Given the fact that the uniform distribution function yields a maximum entropy, Alg. 1 maximizes uniformity in the GFH by normalizing GFH values in histogram \mathcal{H}_{Δ} . The routine constructs an empty histogram $\hat{\mathcal{H}}_{\Delta}$ and iteratively moves points from \mathcal{H}_{Δ} to $\hat{\mathcal{H}}_{\Delta}$. This is done by per-row sampling from \mathcal{H}_{Δ} via cyclic iterative selection, going from greater to lower bins and moving a single point in each of the bins k (if there is any) to the corresponding bin k in $\hat{\mathcal{H}}_{\Delta}$. The primary and secondary keys of sampling from a bin k are

$$\mathbf{p}_k = \arg \max_{\mathbf{p} \in {}^k\mathcal{H}_{\Delta}} \Delta_{\mathbf{p}}, \quad \mathbf{p}_k = \arg \max_{\mathbf{p} \in {}^k\mathcal{H}_{\Delta}} \|\mathbf{p}\|_2. \quad (33)$$

RMS does not balance rotation-space observability but exploits the fact that the rotational rate of residuals is a function of $\|\mathbf{p}\|_2$, as defined in Rem. 3. This is done via the secondary key in Eq. (33), which values points by their potential for being part of a large-magnitude residual in the later correspondence-matching part of the estimation. Note that when the assumption of small rotations in Rem. 3 is not met, the invariance to rotations no more applies, leading to suboptimal sampling.

The iterative sampling process is terminated once the termination criteria in Eq. (27) is satisfied and $i \geq K$. Since the entropy reaches its maximum at K steps, the maximum entropy rate $\bar{\mu}_{\Delta}^*$ is guaranteed to be found at K steps at maximum. The $i \geq K$ condition thus allows redefining Eq. (31) as

$$\bar{\mu}_{\Delta}^*(\mathcal{P}_v) = \max_{i=\{1, \dots, K\}} \left(\bar{\mu}_{\Delta}({}^i\hat{\mathcal{P}}) \right), \quad (34)$$

where ${}^i\hat{\mathcal{P}}$ is the sampled point-set at iteration i . After terminating the routine at iteration $i \geq K$, the sampled points $\hat{\mathcal{P}} = {}^i\hat{\mathcal{P}}$ equal to all the points sampled up to iteration i .

This entropy-maximizing approach normalizes the spectrum of Δ , and thus introduces a certain level of redundancy defined in Sec. IV-A (by including points with low Δ). A certain level of redundancy helps in maintaining the original spatial distribution of the points (similar to voxelization), which is beneficial in reducing the effects of noise and outliers. It has been verified experimentally that the entropy maximization of GFH is more resilient than maximizing the cumulative sum of GFH, which tends to under-constrain the problem and is sensitive to noise and outliers.

Algorithm 1 Information-maximizing point selection

```

1: Input:
2:  $\mathcal{P} = \{\mathbf{p}\}, \mathbf{p} \in \mathbb{R}^3$  ▷ input point set
3:  $v \in \mathbb{R}^+$  ▷ voxel size in meters
4:  $K \in \mathbb{Z}^+$  ▷ number of histogram bins
5:  $\lambda_{\Omega} \in (0, 1)$  ▷ entropy-rate termination criteria (Eq. (27))
6:  $C, R \in \mathbb{Z}^+$  ▷ number of columns and rows (if  $\mathcal{P}$  in matrix form)
7:  $\theta_v \in \mathbb{R}^+$  ▷ vertical field of view of the sensor (if  $\mathcal{P}$  in matrix form)
8: Output:
9:  $\hat{\mathcal{P}} \subseteq \mathcal{P}$  ▷ point subset maximizing GFH entropy, Eq. (27)
10: Begin:
11:  $\mathcal{P}_v = \text{voxelize}(\mathcal{P}, v)$ 
12:  $\mathbb{K}_v = \text{KDTree}(\mathcal{P}_v)$  ▷ construct KD-tree for efficient NN search
13:  $\Delta_{\mathcal{P}_v} = \text{GFH}(\mathcal{P}_v, \mathbb{K}_v, C, R, \theta_v)$  ▷ Eq. (23)-(25)
14:  $\bar{\Delta}_{\mathcal{P}_v} = \text{normalizeGFH}(\Delta_{\mathcal{P}_v})$  ▷ Eq. (32)
15:  $\blacktriangleright$  Construct a histogram of GFH values
16:  $\mathcal{H}_{\Delta} = \text{histogram}(\bar{\Delta}_{\mathcal{P}_v}, K)$  ▷ discretize  $\bar{\Delta}_{\mathcal{P}_v}$  into  $K$  fixed-sized bins
17:  $\hat{\mathcal{H}}_{\Delta} = \text{histogram}(\emptyset, K)$  ▷ empty histogram of  $K$  fixed-sized bins
18:  $\blacktriangleright$  Compute maximum entropy rate  $\bar{\mu}_{\Delta}^*$ 
19:  $\bar{\mu}_{\Delta}^* = 0$ 
20: for each  $k \in \langle 1, K \rangle$  do ▷ iterate each bin exactly once
21:    ${}^k\mathcal{H}_{\Delta} = \text{sort}({}^k\mathcal{H}_{\Delta})$  ▷ sort bin  $k$  in desc. order by Eq. (33)
22:    ${}^k\hat{\mathcal{H}}_{\Delta} = {}^k\mathcal{H}_{\Delta} \cup \{{}^k\mathcal{H}_{\Delta}.\text{pop}()\}$  ▷ move highest-value point between bins  $k$ 
23:    $\bar{\mu}_{\Delta}^* = \max\{\bar{\mu}_{\Delta}^*, \bar{\mu}_{\Delta}({}^k\mathcal{H}_{\Delta})\}$  ▷ Eq. (34)
24:  $\blacktriangleright$  Entropy-maximizing selection
25:  $k = K$  ▷ current bin-lookup index
26: while  $|\mathcal{H}_{\Delta}| > 0$  and  $\bar{\mu}_{\Delta}(\hat{\mathcal{H}}_{\Delta}) / \bar{\mu}_{\Delta}^* > \lambda_{\Omega}$  do ▷ terminating via Eq. (27) and (30)
27:    ${}^k\hat{\mathcal{H}}_{\Delta} = {}^k\hat{\mathcal{H}}_{\Delta} \cup \{{}^k\hat{\mathcal{H}}_{\Delta}.\text{pop}()\}$  ▷ move first point in bin  $k$ 
28:    $k = k - 1$  if  $k > 1$  else  $K$  ▷ cyclic right-left iteration
29:  $\hat{\mathcal{P}} = \bigcup_{k \in \langle 1, K \rangle} {}^k\hat{\mathcal{H}}_{\Delta}$  ▷ extract all selected points

```

V. EXPERIMENTAL ANALYSES

Let us compare the proposed approach with three state-of-the-art point cloud sampling methods:

- **V●**: uniform sampling¹ with voxel size $v = \bullet$ cm,
- **NS●**: normal-space voxelization [11] with angular resolution $\pi = \bullet^\circ$ in both azimuth and elevation, and
- **CovS●**: covariance-based sampling² [12] with sampled-to-all point ratio of $\rho = \bullet\%$.

All the state-of-the-art methods and the proposed approach were integrated into two state-of-the-art odometry (no loop closures) pipelines: KISS-ICP [7] and LOAM [8]. KISS-ICP is a state-of-the-art implementation of the ICP algorithm, a typical case of a *dense* method utilizing the *point-to-point* metric. LOAM is a *feature*-based odometry method extracting plane and line features. LOAM represents a basis for the majority of the *feature*-based state-of-the-art methods. Since the proposed sampling method is algorithm-independent, it has the potential for positively improving all the other LiDAR-based odometry and SLAM methods building upon ICP and LOAM algorithms. To remain close to the core principles and to reduce the effects of any additional concepts, these two

¹Open-source implementation taken from KISS-ICP [7].

²Open-source implementation taken from PointMatcher [26].

representative odometry pipelines have been chosen for their minimalism on purpose.

To ensure a fair comparison, the best parametrizations balancing convergence and real-time processing were fine-tuned manually for all methods, both odometry pipelines, and all datasets. These parametrizations are given in Tab. IV. All the experiments were performed on AMD Ryzen 7 PRO 4750U (comparable performance verified on Intel® Core i7-10710U).

A. Datasets

The datasets used in evaluation are summarized in Tab. I. Their selection includes custom data from UAVs (D1-D3) covering full six-DoF movements in different degraded contexts, and KITTI (D4) and Hilti-Oxford (D5) as two of the most prevalent datasets used in evaluating LiDAR-based methods in the related literature. Only 3D LiDAR data are used.

TABLE I: Table of used datasets.

ID	Dataset	Work	Platform	Real world	Point count
D1	X-ICP	[4]	Drone	✗	64×1024 @ 10 Hz
D2	Dronument	[6] ³	Drone	✓	16×1024 @ 10 Hz
D3	DARPA SubT	[27] ³	Drone	✓	64×512 @ 10 Hz
D4	KITTI	[5]	Car	✓	16×1024 @ 10 Hz
D5	Hilti-Oxford	[28]	Handheld	✓	32×2000 @ 10 Hz

B. Parametrization of RMS

It has been validated empirically that out of the three sensor-agnostic parameters in Alg. 1, K and ν have limited effect on the performance. Thus, $K = 10$ remains fixed in all the presented experiments and ν is selected such that uniform sampling $\mathbf{V}\bullet$ is the most accurate and computes in real time in the given dataset. Tab. II presents an ablation study on the maximum relative entropy rate $\lambda_{\bar{H}}$. The table demonstrates that the algorithm is stable once $\lambda_{\bar{H}}$ lies in a reasonable interval, here 0.2-0.7%. Based on Tab. II, we use $\lambda_{\bar{H}} = 0.4\%$ in all our KISS-ICP [7] experiments as a balance between runtime, accuracy, and stability. Since the stable interval is pipeline-dependent, similar grid-search has been done to find optimal $\lambda_{\bar{H}}$ for the *feature*-based LOAM [8] estimation pipeline used in Sec. V-D. In LOAM, one instance of Alg. 1 runs independently for each of the feature types, with $\lambda_{\bar{H}}$ being fixed to $\lambda_{\bar{H}} = 0.8\%$ for plane and $\lambda_{\bar{H}} = 15\%$ for line features.

TABLE II: Influence of the maximum relative entropy rate $\lambda_{\bar{H}}$ on performance of the proposed method in experiment presented in Fig. 5.

$\lambda_{\bar{H}}(\%)$	0.1	0.2	0.3	0.4	0.5	0.7	1.0
RMSE (m)	0.31	0.22	0.25	0.24	0.27	0.29	0.43
avg. time (ms)	42	37	32	29	27	24	23
compr. rate (%)	95.3	97.0	97.8	98.2	98.5	98.9	99.2

C. Convergence Analysis

Fig. 5 demonstrates an experiment designed to compare performance of the four sampling techniques (all fine-tuned to the environment). In the experiment, a UAV performs a loop inside a challenging simulation world (D1) designed to contain various geometrical degeneracies (translational along narrow corridors and rotational within a circular room). The experiment shows that the proposed method outperforms the baseline methods in terms of speed, accuracy, and robustness (even to geometrical degeneracies), all while sampling the

³Dataset available at github.com/ctu-mrs/slam_datasets.

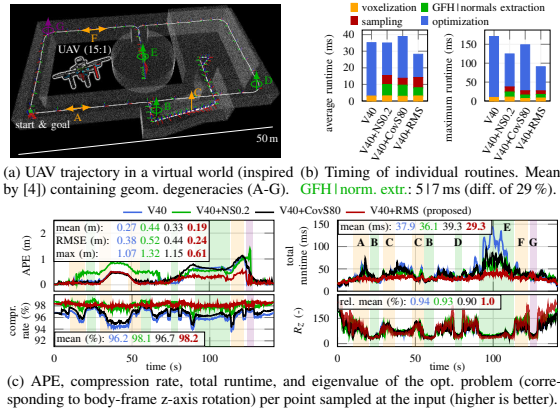


Fig. 5: Output of KISS-ICP [7] 6-DoF odometry when preceded by different point cloud sampling methods. Parametrization: best performing for each method, robot: multirotor UAV, sensor range: 30 m, sensor noise: none. (a) Shows areas of translational (A, C, F) and body-frame z-axis rotational (B, D, E, G) degeneracy. At D and G, the degeneracy arises (see low values of the R_z eigenvalue) from large UAV tilt, which orients the LiDAR such that its data are degenerate around the z-axis. At G, a "loop closing" emerges naturally (see APE). (b) Due to the high compression rate and by balancing the translational space, RMS samples points such that they yield the fastest optimization convergence. (c) RMS yields the lowest drift, removes the largest amount of points, produces stable and lowest runtime, and preserves the highest information rate for optimization (only R_z shown).

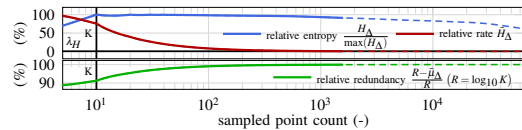


Fig. 6: Rel. entropy, rel. entropy rate, and rel. redundancy of GFH at time 60s of experiment in Fig. 5. $\mathbf{V40+RMS}$ sampled about 2% of points out of 65k total. Dashed lines represent the non-sampled points.

least amount of points. The data show superior timing and compression consistency of the proposed method, with both reaching almost constant values with a limited number of outliers. This is a particularly important attribute for deployment of small and agile resource-constrained robots with real-time constraints, such as UAVs. Fig. 5c shows that information rate (measured as eigenvalue per point sampled on input) extracted in the optimization from the problem Hessian is highest in RMS (only z-axis rotational eigenvalue R_z is shown). Additionally, Fig. 6 showcases a single-frame sampling of Alg. 1.

D. Quantitative analysis

Tab. III presents a quantitative analysis comparing the effects of the sampling methods on the two odometry pipelines. Together with Tab. IV, the two tables show that the fixed parametrization adapts well to various different sensors, environments, and conditions. This is a significant practical advantage, which improves the method's applicability by reducing the need for tuning the proposed method to every domain. The data show superior performance of the proposed method in

- improving performance in well-conditioned settings,
- reducing odometry drift in degenerated conditions,
- sampling the least amount of points in general, and
- computing the fastest while being the most accurate.

TABLE III: Quantitative performance of KISS-ICP and LOAM pose estimation pipelines when preceded by four different 3D LiDAR point cloud sampling techniques: uniform sampling $V\bullet$, normal-space sampling $V\bullet+NS\bullet$ [11], covariance sampling $V\bullet+Cov\bullet$ [12], and redundancy-minimizing sampling $V\bullet+RMS$ (proposed). Metrics: APE $|\delta|$ (m), RPE $\Delta\delta$ (m), total runtime τ (ms), and compression rate χ (%). The best results are in **bold**. Trajectories of experiments D2 and D4: KITT1 #00 are shown in Fig. 1.

Dataset	Metric	KISS-ICP				LOAM				
		$V\bullet$	$V40+NS\bullet$	$V40+Cov\bullet$	$V40+RMS$	$V\bullet$	$V\bullet+NS\bullet$	$V\bullet+Cov\bullet$	$V\bullet+RMS$	
D2: Stará Voda #00	$ \delta _{rms}$					0.85	0.31	0.19	0.12	
	$ \delta _{mean}$					0.39	0.18	0.15	0.09	
	$ \delta _{max}$					2.56	0.91	0.45	0.39	
	$\Delta\delta_{rms}$	failed in estimating vertical motion				0.06	0.02	0.01	0.02	
	$\Delta\delta_{mean}$	failed in estimating vertical motion				0.01	0.01	0.01	0.01	
	$\Delta\delta_{max}$	failed in estimating vertical motion				1.57	0.39	0.12	0.18	
	τ_{mean}					69.0	61.4	37.9	31.0	
	τ_{max}					109.6	138.8	74.3	69.7	
	χ_{mean}					74.8	82.0	90.9	95.7	
	D3: urban corridor	$ \delta _{rms}$	0.59	0.63	0.67	0.42	1.20	2.01	0.75	1.20
		$ \delta _{mean}$	0.50	0.52	0.63	0.38	1.14	1.78	0.69	1.16
		$ \delta _{max}$	1.24	1.34	1.11	0.86	1.84	4.00	1.41	1.78
		$\Delta\delta_{rms}$	0.04	0.04	0.04	0.04	0.03	0.04	0.03	0.03
		$\Delta\delta_{mean}$	0.03	0.03	0.04	0.03	0.02	0.03	0.03	0.02
$\Delta\delta_{max}$		0.23	0.29	0.24	0.23	0.12	0.18	0.17	0.12	
τ_{mean}		13.5	18.1	24.1	16.7	13.0	15.1	11.0	13.0	
τ_{max}		61.3	56.8	77.9	55.7	36.3	32.7	24.9	28.6	
χ_{mean}		95.8	96.2	93.4	95.9	94.8	99.0	99.1	98.5	
D4: KITT1 #00		$ \delta _{rms}$	248.60	14.31	56.01	8.35	20.72	25.11	12.28	12.76
		$ \delta _{mean}$	205.97	11.27	45.29	7.48	16.34	21.05	9.50	9.78
		$ \delta _{max}$	458.96	32.59	116.34	16.20	49.87	51.82	29.96	29.45
		$\Delta\delta_{rms}$	1.41	1.27	1.27	1.27	1.27	1.27	1.27	1.27
		$\Delta\delta_{mean}$	1.26	1.17	1.17	1.17	1.17	1.17	1.17	1.17
	$\Delta\delta_{max}$	14.86	14.71	14.64	14.67	14.79	14.72	14.70	14.69	
	τ_{mean}	43.3	57.1	34.2	35.7	83.0	76.3	74.9	66.9	
	τ_{max}	769.1	270.8	88.4	87.7	180.8	196.9	168.1	159.6	
	χ_{mean}	96.6	97.2	99.2	99.3	89.4	99.2	99.1	99.4	
	D4: KITT1 #09	$ \delta _{rms}$	481.02	17.48	25.66	15.76	16.08	21.15	12.64	10.77
		$ \delta _{mean}$	418.98	13.89	19.31	13.12	11.94	15.72	9.18	8.07
		$ \delta _{max}$	772.04	46.24	68.67	32.61	38.86	50.01	31.49	25.59
		$\Delta\delta_{rms}$	2.19	1.65	1.75	1.58	1.58	1.58	1.58	1.58
		$\Delta\delta_{mean}$	1.71	1.57	1.62	1.52	1.52	1.52	1.52	1.52
$\Delta\delta_{max}$		25.22	4.28	7.72	3.58	3.60	3.62	3.62	3.65	
τ_{mean}		71.3	67.0	59.8	42.8	70.2	59.1	59.33	50.8	
τ_{max}		1748.3	811.3	700.5	118.0	143.4	119.2	127.3	92.7	
χ_{mean}		95.9	96.4	97.1	98.8	87.1	99.0	98.5	99.4	
D5: Hilti-Oxford #04		$ \delta _{rms}$					0.34	0.63	0.29	0.23
		$ \delta _{mean}$					0.30	0.59	0.26	0.21
		$ \delta _{max}$					0.74	1.08	0.67	0.58
		$\Delta\delta_{rms}$	failed in estimating quick rotational motions				0.14	0.14	0.14	0.14
		$\Delta\delta_{mean}$	failed in estimating quick rotational motions				0.10	0.10	0.10	0.10
	$\Delta\delta_{max}$	failed in estimating quick rotational motions				0.56	0.61	0.54	0.51	
	τ_{mean}					67.9	55.7	54.1	42.8	
	τ_{max}					138.5	133.9	120.0	77.1	
	χ_{mean}					90.5	95.4	95.3	97.7	
	D5: Hilti-Oxford #14	$ \delta _{rms}$					2.22	2.85	0.84	0.76
		$ \delta _{mean}$					1.86	2.35	0.62	0.62
		$ \delta _{max}$					3.69	4.82	2.06	1.64
		$\Delta\delta_{rms}$	failed in estimating quick rotational motions				0.16	0.16	0.16	0.16
		$\Delta\delta_{mean}$	failed in estimating quick rotational motions				0.09	0.09	0.09	0.09
$\Delta\delta_{max}$		failed in estimating quick rotational motions				1.91	1.91	1.91	1.91	
τ_{mean}						20.7	18.2	17.3	17.2	
τ_{max}						45.1	47.6	40.4	30.3	
χ_{mean}						97.9	98.7	98.7	98.8	

The reported timings are for the entire pipeline, including point or feature sampling, and the optimization. The accuracy gains are associated with lower comp. time enabling use of all data in real-time as well as with high noise and outlier removal.

REFERENCES

- [1] J. Zhang *et al.*, “On Degeneracy of Optimization-Based State Estimation Problems,” in *IEEE ICRA*, 2016, pp. 809–816.
- [2] J. Jiao *et al.*, “Greedy-Based Feature Selection for Efficient LiDAR SLAM,” in *IEEE ICRA*, 2021, pp. 5222–5228.
- [3] W. Li *et al.*, “KFS-LIO: Key-Feature Selection for Lightweight Lidar Inertial Odometry,” in *IEEE ICRA*, 2021, pp. 5042–5048.

TABLE IV: Table of parameters used in the experiments in Tab. III. The values marked with * are for the $V\bullet$ method only (the $V40$ parametrizations use $v = 40$ cm). Among the compared methods, only the proposed method’s parametrization is fixed (λ_R), showing the method’s unique adaptability to different sensors and environments.

Pipeline Sampling	KISS-ICP 3D points				LOAM plane features				LOAM line features				
	Parameter Units	v (m)	π (°)	ρ (%)	λ_{ij} (%)	v (m)	π (°)	ρ (%)	λ_{ij} (%)	v (m)	π (°)	ρ (%)	λ_{ij} (%)
D2		*x	*x	*x	*x	0.4	0.2	40		0.2	0.2	20	
D3		0.4	0.2	50		0.8	0.2	60		0.4	0.2	30	
D4: #00		1.0*	1.0	10	0.4	0.4	0.1	80	0.8	0.2	0.1	20	15
D4: #09		1.0*	1.0	30		1.0	0.1	50		0.5	0.1	2.5	
D5: #04		*x	*x	*x	*x	0.4	1.0	50		0.2	1.0	50	
D5: #14		*x	*x	*x	*x	0.4	1.0	80		0.2	1.0	2.5	

- [4] T. Tuna *et al.*, “X-ICP: Localizability-Aware LiDAR Registration for Robust Localization in Extreme Environments,” *IEEE Transactions on Robotics*, vol. 40, pp. 452–471, 2024.
- [5] A. Geiger *et al.*, “Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite,” in *IEEE/CVF CVPR*, 2012, pp. 3354–3361.
- [6] P. Petracek *et al.*, “New Era in Cultural Heritage Preservation: Cooperative Aerial Autonomy for Fast Digitalization of Difficult-to-Access Interiors of Historical Monuments,” *IEEE RA-M*, pp. 2–19, 2023.
- [7] I. Vizzo *et al.*, “KISS-ICP: In Defense of Point-to-Point ICP textandash Simple, Accurate, and Robust Registration If Done the Right Way,” *IEEE RA-L*, vol. 8, no. 2, pp. 1029–1036, 2023.
- [8] J. Zhang *et al.*, “LOAM: Lidar Odometry and Mapping in Real-time,” in *Robotics: Science and Systems*, 2014.
- [9] C. R. Qi *et al.*, “PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space,” in *NIPS*, vol. 30, 2017.
- [10] R. B. Rusu *et al.*, “3D Is Here: Point Cloud Library (PCL),” in *IEEE ICRA*, 2011, pp. 1–4.
- [11] S. Rusinkiewicz *et al.*, “Efficient Variants of the ICP Algorithm,” in *3DIM*, 2001, pp. 145–152.
- [12] N. Gelfand *et al.*, “Geometrically Stable Sampling for the ICP Algorithm,” in *3DIM*, 2003, pp. 260–267.
- [13] I. Lang *et al.*, “SampleNet: Differentiable Point Cloud Sampling,” in *IEEE/CVF CVPR*, 2020, pp. 7575–7585.
- [14] E. Nezhadarya *et al.*, “Adaptive Hierarchical Down-Sampling for Point Cloud Classification,” in *IEEE/CVF CVPR*, 2020, pp. 12953–12961.
- [15] Y. Wang *et al.*, “Dynamic Graph CNN for Learning on Point Clouds,” *ACM Transactions on Graphics*, vol. 38, no. 5, pp. 146:1–146:12, 2019.
- [16] Y. Yang *et al.*, “FoldingNet: Point Cloud Auto-Encoder via Deep Grid Deformation,” in *IEEE/CVF CVPR*, 2018, pp. 206–215.
- [17] Y. Shen *et al.*, “Mining Point Cloud Local Structures by Kernel Correlation and Graph Pooling,” in *IEEE/CVF CVPR*, 2018, pp. 4548–4557.
- [18] T. Ozaslan *et al.*, “Autonomous Navigation and Mapping for Inspection of Penstocks and Tunnels With MAVs,” *IEEE RA-L*, vol. 2, no. 3, pp. 1740–1747, 2017.
- [19] M. Tranzatto *et al.*, “CERBERUS in the DARPA Subterranean Challenge,” *Science Robotics*, vol. 7, no. 66, p. eabp9742, 2022.
- [20] R. Schnabel *et al.*, “Octree-Based Point-Cloud Compression,” in *IEEE VGTC Conference on Point-Based Graphics*, 2006, pp. 111–121.
- [21] Y. Duan *et al.*, “PFilter: Building Persistent Maps through Feature Filtering for Fast and Accurate LiDAR-based SLAM,” in *IEEE/RSJ IROS*, 2022, pp. 11 087–11 093.
- [22] Z. Zhou *et al.*, “ROI-cloud: A Key Region Extraction Method for LiDAR Odometry and Localization,” in *IEEE ICRA*, 2020, pp. 3312–3318.
- [23] H. Lim *et al.*, “A Single Correspondence Is Enough: Robust Global Registration to Avoid Degeneracy in Urban Environments,” in *IEEE ICRA*, 2022, pp. 8010–8017.
- [24] F. Li *et al.*, “Reducing Redundancy in Maps without Lowering Accuracy: A Geometric Feature Fusion Approach for Simultaneous Localization and Mapping,” *JGI*, vol. 12, no. 6, p. 235, 2023.
- [25] H. Pottmann *et al.*, “Geometry of the Squared Distance Function to Curves and Surfaces,” in *Visualization and Mathematics III*, G. Farin *et al.*, Eds., 2003, pp. 221–242.
- [26] F. c. Pomerleau *et al.*, “Comparing ICP Variants on Real-World Data Sets,” *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, 2013.
- [27] M. Petrlík *et al.*, “UAVs Beneath the Surface: Cooperative Autonomy for Subterranean Search and Rescue in DARPA SubT,” *Field Robotics*, vol. 3, no. 1, pp. 1–68, 2023.
- [28] L. Zhang *et al.*, “Hilti-Oxford Dataset: A Millimeter-Accurate Benchmark for Simultaneous Localization and Mapping,” *IEEE RA-L*, vol. 8, no. 1, pp. 408–415, 2023.

- [2c] **P. Petracek**, V. Kratky, T. Baca, M. Petrlik, and M. Saska, “New Era in Cultural Heritage Preservation: Cooperative Aerial Autonomy for Fast Digitalization of Difficult-to-Access Interiors of Historical Monuments,” *IEEE Robotics and Automation Magazine*, pp. 2–19, 2023. **Equal contribution of the first two co-authors.**

Our interdisciplinary outreach into the field of restoration and cultural preservation began initially in [1c, 9a, 131], followed by [10a, 132], and culminated into a core publication summarizing the research in documenting large interiors by autonomous UAV teams [2c]. The work, developed in close cooperation with the Czech National Heritage Institute, was published in the IEEE Robotics and Automation Magazine and presented physically at IEEE IROS 2023. Although targeting a general public audience, this work proposed a principal three-phased robotic methodology to the task of documenting large historical monuments with cooperating autonomous aerial robots, as visualized in Figure 3.2. Nevertheless, the developed methodology is not limited to historical monuments and is transferable into general interior environments, including industrial plants. Among the contributions is the LiDAR-based localization and mapping pipeline suited for UAVs, which has been in development since [10a]. The methodology employs an on-board 3D LiDAR as the main source of information used in estimating the UAV state with a LiDAR odometry. It also utilizes a sparse ground-based map of the environment, which is easily available or obtainable with a terrestrial LiDAR scanner. However, in contrast to [1c], the map is not used in localization but is used for three different other reasons. First, the map is used in the proposed coarse-to-fine initial alignment of the UAVs within the interior in order to obtain a global reference frame used in accurate UAV synchronization and the targeting of objects of documentation interest. This allows the UAVs to be deployed within a shared environment and effectively coordinate within the scope of the documentation mission. Second, it serves as a baseline, allowing one to specify, plan, visualize, and simulate the optimal documentation mission. Last and most importantly, the map is used as a safety and robustness measure, allowing the quantification of drift of onboard pose estimation and thus detecting the presence and effects of emerging geometrical degeneracy. Since the targeted missions inside historical monuments require a critical level of safety, the deployments in this task are halted when geometrical degeneracy negatively influences a robot’s pose estimation.

In addition, the outcomes of [2c] extend beyond the field of robotics by contributing to the fields of historical conservation and cultural preservation. The outcomes of [2c] are listed in detail in Section 5.3.

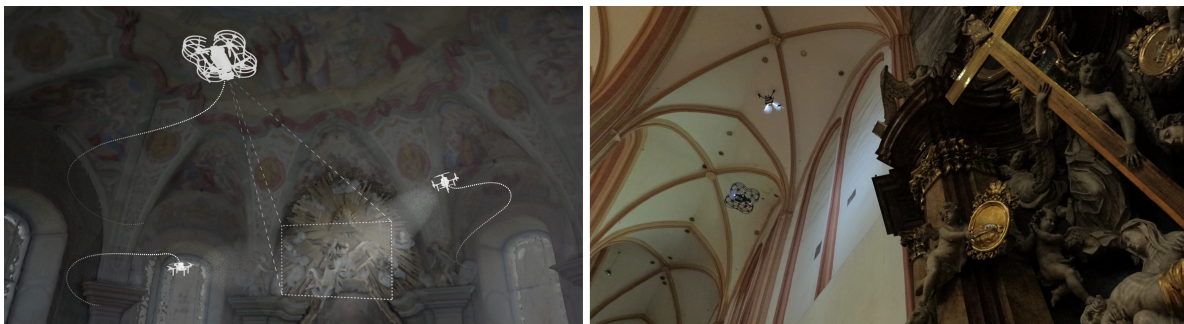


Figure 3.2: Virtual (left) and real-world (right) documentation of a historical monument by a synchronized team of autonomous UAVs, with one UAV taking high-resolution photographs whilst the rest highlight details in the documented scene by serving as dynamic lighting.

New Era in Cultural Heritage Preservation

Cooperative Aerial Autonomy for Fast Digitalization of Difficult-to-Access Interiors of Historical Monuments

Pavel Petracek^{*}, Vit Kratky^{*}, Tomas Baca^{*}, Matej Petrlik^{*}, and Martin Saska^{*}

Abstract—Digital documentation of large interiors of historical buildings is an exhausting task since most of the areas of interest are beyond typical human reach. We advocate the use of autonomous teams of multi-rotor Unmanned Aerial Vehicles (UAVs) to speed up the documentation process by several orders of magnitude while allowing for a repeatable, accurate, and condition-independent solution capable of precise collision-free operation at great heights. The proposed multi-robot approach allows for performing tasks requiring dynamic scene illumination in large-scale real-world scenarios, a process previously applicable only in small-scale laboratory-like conditions. Extensive experimental analyses range from single-UAV imaging to specialized lighting techniques requiring accurate coordination of multiple UAV. The system’s robustness is demonstrated in more than two hundred autonomous flights in fifteen historical monuments requiring superior safety while lacking access to external localization. This unique experimental campaign, cooperated with restorers and conservators, brought numerous lessons transferable to other safety-critical robotic missions in documentation and inspection tasks.

I. AUTONOMOUS AERIAL ROBOTICS FOR HERITAGE DIGITALIZATION

Digital documentation of large interiors of historical buildings is an exhausting task since most of the areas of interest are beyond typical human reach. We advocate the use of fully-autonomous teams of cooperating multi-rotor Unmanned Aerial Vehicles (UAVs) to speed up the documentation process by several orders of magnitude while allowing for a repeatable, accurate, and condition-independent solution capable of precise collision-free operation at great heights. In particular, we present a universal autonomy for UAVs cooperating aurally within a team while documenting the interiors of historical buildings for the purposes of restoration planning and documentation works, as well as for assessing the structural state of aging historical sites. We show that the proposed approach of active multi-robot cooperation enables performing documentation tasks requiring dynamic scene illumination in large-scale real-world scenarios, a process previously applicable only manually in areas easily accessible by humans.

The presented system was developed in cooperation with cultural heritage institutions as part of the Dronument project [1] and was deployed fully autonomously in numerous characteristically diverse historical monuments, as exhibited in Fig. 1 and Table III. The included experimental evaluation utilizes UAVs in multiple real-world documentation tasks, and

discusses the quality of the obtained results used in subsequent restoration works, as well as suitability of particular techniques for UAVs. The analyses demonstrate the framework’s robustness in single and multi-robot deployments in more than two hundred fully-autonomous flights in fifteen historical monuments. In these experiments, the aerial robots rely solely on onboard sensors without access to external localization such as global navigation satellite systems (GNSSs) or motion capture systems, which significantly increases deployability of the system. This unique, extensive, experimental campaign, which cooperated with restorers and conservators, brought numerous lessons learned that are transferable to other safety-critical robotic missions in documentation and inspection tasks. The system also serves as a large part of an official methodological study approved by the Czech National Heritage Institute for its high added value in heritage protection. The methodology (available at [1]) describes the proper usage of UAVs in historical structures for the first time and so prescribes the proposed system to be a standard in this application.

II. BACKGROUND

Often serving educational, cultural, or social purpose, the preservation of cultural heritage as a valuable reminder of our history is in the greater interest of society. Cultural management and preservation of historical monuments became a relevant topic in the late 19th and 20th centuries when many valuable historical monuments were destroyed while establishing modern infrastructure. By introducing cultural heritage preservation into legislation, the monuments gained protection from human interference. However, being exposed to real-world conditions continually degrades historical buildings and artifacts within. This has initiated the endeavor to actively prevent the irreversible damage of cultural heritage by monitoring its condition and performing restoration and conservation works.

Conservation work on a historical artifact comprises four consecutive phases: the initial survey, the choice of restoration steps and costs evaluation, the actual restoration works, and continued monitoring of the restoration. Both the initial survey and monitoring phase require providing information about the artifact in digital form (usually camera imaging). Thus, these phases are considered a data collection task for which an aerial vehicle, capable of gathering data in a cost-effective and fast manner, can be of great help. This is especially true for areas of interest which are located beyond typical human reach, a situation often arising in tall historical buildings such as churches and cathedrals. Apart from planning restoration works, gathered digital materials can support the reconstruc-

^{*} Authors are with the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic.
[×] Pavel Petracek and Vit Kratky are co-first authors.
 Corresponding author: pavel.petracek@fel.cvut.cz

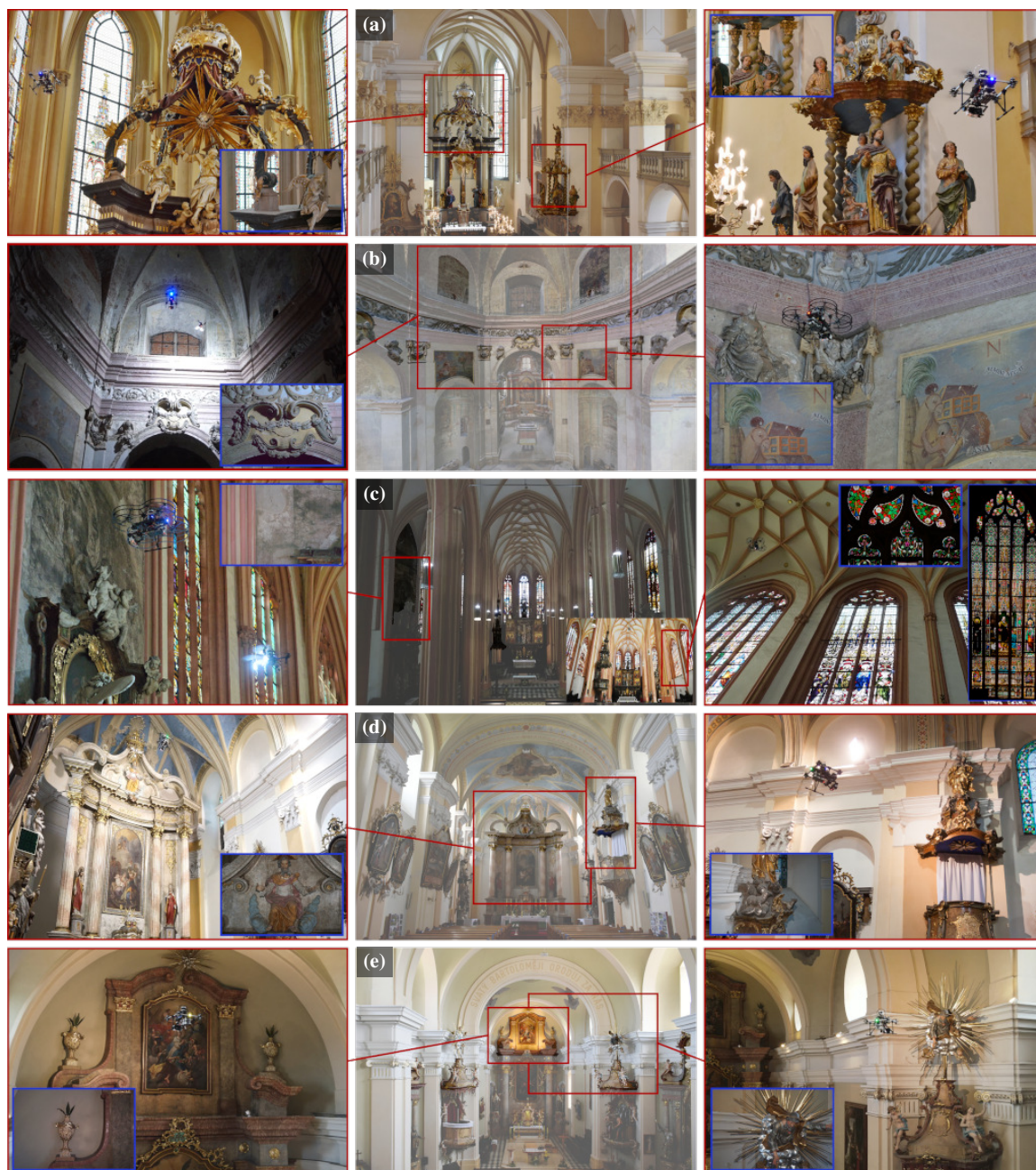


Fig. 1: Illustration of deployment of the presented methodology in selected historical buildings located in the Czech Republic — (a) Church of the Exaltation of the Holy Cross in Prostějov, (b) St. Anne and St. Jacob the Great Church in Stará Voda by Libavá, (c) Church of St. Maurice in Olomouc, (d) Church of the Nativity of the Virgin Mary in Nový Malín, and (e) Church of St. Bartholomew in Zábřeh. Center images show the interiors of the churches with highlighted objects of documentation interest. Side images show actual deployment of UAVs in the particular settings together with example images (highlighted in blue) captured by an onboard camera.

tion of a structure in the event of its sudden accidental destruction (e.g., the burning of the Notre-Dame Cathedral in 2019).

III. ROBOTICS AND AUTOMATION IN CULTURAL HERITAGE PRESERVATION

Documentation and digitalization of historical objects requires gathering various types of data, e.g., camera images in visible, infrared (IR) and ultraviolet (UV) spectra, and 3D models. The data gathering is demanding in both time and human resources, particularly in large buildings. This motivates the endeavor to automate data gathering by introducing mobile robotic solutions capable of fast autonomous documentation. The first level of mobile-robot automation can be achieved by applying Unmanned Ground Vehicles (UGVs) as carriers of the documentation sensors. A UGV equipped with a laser scanner and capable of autonomous navigation in constrained environments can sequentially visit several locations to collect a set of scans covering the entire operational space [2]. An advantage of this approach lies primarily in reducing necessary human participation in the scanning process, allowing for the collection of scans from potentially dangerous areas. Several systems applying such an approach were already developed and deployed for scanning historical monuments [3], [4].

Whereas the operational space of UGVs usually does not exceed typical human reach, multi-rotor UAVs capable of 3D navigation in confined environments can be applied for data collection tasks in difficult-to-access areas. In exteriors, UAV solutions abundantly utilize predefined GNSS poses for navigation [5]. In contrast to exteriors, the applicability of UAVs in interiors imposes additional challenges — lack of GNSS localization, navigation in a confined environment, and non-negligible aerodynamic effects. Because of that, UAV systems deployed for indoor data gathering are mainly limited to industrial inspections, with only a few works targeting UAV-based documentation of historical buildings. The specifics of such an application are targeted in this work.

For industrial inspections, the literature typically exploits the environment structure, such as known profiles of tunnels [6] or structured and well-lit warehouses [7]. More general solutions were introduced in the commercial sector introducing semi-autonomous UAV inspection systems¹— DJI Mavic 3, Elios 3, or Skydio 2+TM. In interiors, DJI provides image-based UAV stabilization, Elios allows for human-operated flight with LiDAR and camera-based stabilization and mapping with guarantees of environmental and mechanical protection, and SkydioTM offers automated camera-stabilized flight for interactive 3D reconstruction. Although all these solutions provide an assistive level of autonomy in UAV stabilization, the first two require human-in-the-loop navigation. None of the mentioned solutions offer full interior autonomy, repeatability, modularity, rotor nor sensory redundancy, imaging focusing on capturing high-quality details, and cooperative multi-robot deployment.

As mentioned, aerial data gathering inside historical buildings is rare. A specialized platform for assisting in cultural heritage monitoring called *HeritageBot* was introduced in [8].

¹DJI Mavic 3: [dji.com/cz/mavic-3](https://www.dji.com/cz/mavic-3), Elios 3: [flyability.com/eli0s-3](https://www.flyability.com/eli0s-3), Skydio 2+TM: [skydio.com/skydio-2-plus](https://www.skydio.com/skydio-2-plus).

However, no evidence of the deployment of this platform in historical monuments is presented. In [9], the authors propose an assistive system to manual control of the UAV during inspection tasks with the experimental deployment of the system inside and outside historical sites.

Among introduced solutions, the most advanced UAV-based systems with the high level of autonomy required for the interiors of historical buildings were introduced in our recent works [10]–[13]. In these publications, we introduced a preliminary application-tailored autonomous UAV system allowing for safe localization and navigation inside historical structures [10], the methodology and algorithms for the realization of advanced documentation techniques found in reflectance transformation imaging (RTI) [11] and raking light (RAK) [12], and an autonomous single-UAV system for realization of documentation missions [13]. All works provide a fully autonomous solution and the possibility of performing documentation techniques in difficult-to-access areas without using mobile lift platforms or scaffolding installation. Here, we progress beyond previous works by introducing a full 3D simultaneous localization and mapping (SLAM) methodology for indoor localization of robots; by advancing robustness to localization drifts and hard-to-detect obstacles with additional sensory redundancy; by improving path, trajectory, and mission planning; by using a UAV team to realize documentation techniques that could not be realized with only a single robot in principle; and by presenting the complete set of results achieved in the Dronument project that are summarized in numerous lessons learned during the unique experimental campaign within highly safety-critical missions.

IV. DOCUMENTATION TECHNIQUES AND ASSOCIATED CONSTRAINTS

The documentation techniques applied in the field of restoration and cultural heritage preservation aim to capture the current state of the object, survey a potential structural or artistic damage, and determine the age, author and possible dimensions of the elements by identifying the materials and techniques that have been used. For this purpose, diverse methods combining conventional photography in the visible spectrum, photography in invisible spectra making use of different reflective properties of materials, specialized lighting techniques applied for revealing structural details, and even invasive methods based on the collection of material samples are applied. In robotic context, all these methods are associated with varying requirements on sensory equipment, amount of cooperation, and external conditions (mainly illuminance). These relations are summarized in Table I, together with the studied documentation techniques.

The most common documentation technique providing initial information about the studied subject is standard visible spectrum photography (VIS). This technique is applicable to all types of studied objects, ranging from flat paintings and frescoes to 3D structures, including statues and altars. Since the documented areas of historical buildings are often dark, the obtained images suffer from insufficient lighting conditions. Hence, the VIS method often requires additional external lighting to locally increase illuminance, allowing for

TABLE I: Recapitulative table of documentation tasks selected as realizable by aerial vehicles in interiors of historical structures. The squared check marks (☑) identify the realizable documentation methods which were experimentally applied in historical structures, as summarized in section VIII. The last column marks methods for which the ambient light is either required (✓), forbidden (✗), or arbitrary (unmarked).

Documentation technique	Realizable by		Required equipment and lighting conditions		
	Single robot	Multiple robots	Onboard camera	Onboard light	Ambient light
Spectral analysis	visible spectrum: photography (VIS)	☑		✓	✓
	transmitography (VISTR)		✓	✓	
	raking light (RAK)	☑	✓	✓	✓
	three point lighting (TPL)		☑	✓	✓
	reflectance transformation imaging (RTI)	☑	☑	✓	✓
	light-induced luminescence (VIVL)	✓		✓	✓
	UV spectrum: reflectography (UVR)	☑			✓
	fluorescent photography (UVF)	☑	✓	✓	✓
	false-color reflectography (UVRFC)	✓			✓
	IR spectrum: reflectography (IRR)	☑			✓
	transmitography (IRRTR)	✓			✓
	fluorescent photography (IRF)	☑	✓	✓	✓
X-ray: false-color reflectography (IRRFC)	✓			✓	
radiography		✓			
Others	3D reconstruction	☑	✓	✓	
	photogrammetry	✓	✓	✓	✓
	environmental monitoring	✓	✓		

the decreased exposure times required to avoid motion blur from instabilities of a multi-rotor vehicle.

Similar to aesthetic photography, light plays a significant role in restoration documentation. Documentation techniques capturing data in the visible spectrum make use of varying lighting intensity and illumination angles to enhance the quality and amount of information that can be derived from the gathered data. The main group of lighting techniques applicable during documentation tasks aims to highlight the 3D characteristics of captured objects, with three point lighting (TPL) being the most routine. TPL illuminates the object with several sources of luminance, each with different intensity and orientation with respect to the camera's optical axis, in order to provide an aesthetically pleasant and realistic view of the 3D object. Another widely used lighting technique is raking light (RAK), which focuses on revealing the surface details of flat objects. While TPL employs several light sources to avoid overshadowed areas, RAK applies a single light as parallel to the scene as possible. The illumination angle in RAK exploits the shadows to highlight the roughness of the surface.

A highly specialized documentation technique used in the field of restoration is the reflectance transformation imaging (RTI) — an image-based rendering method used for obtaining a representation of an image that enables displaying the image under an arbitrary direction of illumination. The necessary inputs of this method include a set of images of an object taken by a static camera, with each image being under illumination from a different but known direction. The captured images and the corresponding lighting vectors are then used for the computation of a polynomial texture map (PTM) representation of the image that enables an interactive illumination and view of the object. Another specialized documentation technique is visible spectrum transmitography (VISTR) which requires a light source to be positioned behind an object of interest

(OoI) to transmit the light through this object. However, this method is mainly applied for canvas paintings and thus is rather impractical for realization by UAVs.

Multiple techniques exploit UV and IR lumination and its effects. While the methods based on the visible light focus on revealing structural characteristics and colors, the UV and IR methods aim primarily to identify the materials and hidden layers of artworks. The use of different spectra allows more precise dating of the paintings, as the glow of pigment combinations are unique to certain periods. The first group of methods applying UV and IR lights is based on capturing the fluorescent light in the visible spectrum emitted by an object after absorbing UV or IR radiation energy. These methods are called UV fluorescent photography (UVF) and IR fluorescent photography (IRF) and are used for, e.g., detecting zinc and titanium white (UVF) or cadmium red and Egyptian blue (IRF). The second group of methods applying UV and IR lights captures the reflected light in the corresponding spectra. These methods are called UV reflectography (UVR) and IR reflectography (IRR) and are applicable for, e.g., detecting restored areas, highlighting repairs and re-touchings, enhancing faded paintings (UVR) or reaching the underdrawing layer of paintings (IRR).

Except for VIS, all the above-mentioned methods require positioning the light at a certain angle with respect to the camera. Therefore, these methods are not fully realizable by a single UAV and require a multi-robot coordination. The particular methods can be realized in three different configurations dependent on the requirements of the task. The first configuration employs an autonomous multi-robot team consisting of a UAV carrying a documentation sensor and a set of supporting UAVs providing dynamic lighting of the documented scene. The second configuration applies the UAV as a carrier of the sensor whilst the light is provided by external sources. The third

TABLE II: Typical exposure times of the selected documentation techniques.

Technique	Spectrum	Exposure time (s)
visible spectrum photography	visible	≤ 0.2
raking light	visible	≤ 0.2
three point lighting	visible	≤ 0.2
reflectance transformation imaging	visible	≤ 0.2
UV fluorescent photography	UV	≤ 2.0
UV reflectography	UV	2.0
visible spectrum transmitography	visible	2.0
IR reflectography	IR	4.0
IR transmitography	IR	20.0
light-induced luminescence	visible	25.0
IR fluorescent photography	IR	30.0
radiography	X-ray	≥ 30.0

configuration uses the UAV for positioning the light whereas the data are captured by a static sensor from the ground.

The largest problem in realization of the techniques relying on a UAV carrying a camera is the exposure time required for sharp and detailed imaging. Table II summarizes that the exposure times for some of the methods reach tens of seconds. With constraints on image sharpness, such long times and natural *nonstaticity* of highly dynamical multi-rotor UAVs prevent the realization of these techniques in the camera-carrier mode with satisfactory results. Instead, imaging with a static camera and aerial lighting was investigated for some of these techniques.

The non-spectral tasks applied in the field of preservation mostly focus on the 3D reconstruction and environment monitoring through static sensors measuring physical quantities (e.g., temperature, humidity). The most common techniques applied in 3D reconstruction use visible spectrum images (photogrammetry) or scans produced by laser sensors. From the perspective of the proposed system, the data gathering process for 3D reconstruction does not differ from the realization of VIS and collection of raw data from onboard sensors used for localization and mapping. Monitoring the physical quantities in an environment requires attaching a sensor to the UAV frame and navigating it to the required area. If the measurement process requires permanent monitoring, the sensor must be attached at a specific position in the environment (e.g., adhered to a wall or placed on a mantel). This process is also realizable by UAVs but requires fine control, state estimation, and a mechanism for physical robot-to-environment interaction, as closely tackled in [12].

V. UAV-BASED FRAMEWORK FOR DOCUMENTATION OF CULTURAL HERITAGE INTERIORS

The overall pipeline of the UAV-based framework for interior documentation in historical monuments is showcased in Fig. 2. The framework is composed of three main phases — the pre-deployment phase incorporating pre-flight data gathering and mission planning, the actual deployment of the system in interiors of historical buildings, and post-deployment phase, including processing and utilization of the collected data.

A. Pre-deployment Phase

The first step preceding the entire documentation process is obtaining a model of the environment used for safe navigation of the UAV, as well as for the specification of OoFs that should be scanned during documentation missions. For this purpose, a precise terrestrial 3D scanner Leica BLK360 is employed to obtain a set of scans that are later used for building a complete 3D representation of the target environment, both in form of a global point cloud and a 3D model with a colored texture. The colored 3D model serves for precise specification of the desired camera viewpoints and for presenting the documentation outputs to the public and the end users. The camera viewpoints specifications are made by experts of restoration or historical science who position a virtual camera within the 3D model of the environment using a viewpoint-selection tool shown in Fig. 3a. This tool shows a camera and its view and enables to save the camera viewpoint pose in the global coordinate frame. The optical properties of the camera can be parameterized with respect to the equipment available for real-world documentation, thus allowing for visualizing the desired photo to be captured from a given pose in the colored 3D model.

Given the point cloud representation of the environment and the set of to-be-captured images represented by the respective camera viewpoints in the global coordinate frame, the documentation mission plan is generated as follows. First, the problem of finding an optimal sequence σ^* of camera poses minimizing the overall traveled distance is defined and solved as the Traveling Salesman Problem (TSP). Considering the possible dimensionality of the problem, a solver using an efficient Lin-Kernighan heuristics [14] is employed for the solution of TSP to enable on-site plan generation. Constrained by available computational time, the mutual distance between particular pairs of poses within the solution of TSP are given either by the Euclidean distance or by the length of the collision-free path between the poses. Second, the consequent poses in σ^* are connected by the collision-free paths generated with the use of a grid-based planner [15]. This process creates a path connecting all the poses which can be generally unfeasible if limited flight time of a UAV is taken into account. Hence, the final set of plans $\mathbb{P} = \{\mathbf{P}_1, \mathbf{P}_2, \dots, \mathbf{P}_n\}$ is obtained by splitting σ^* to a set of subsequences $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_n\}$, where $\sigma^* = \sigma_1 \cup \sigma_2 \cup \dots \cup \sigma_n$, $\mathbf{P}_i \in \mathbb{P}$ is a collision free path connecting the initial pose with a sequence of poses in σ_i , and equation $t(\mathbf{P}_i) < t_{max}$ holds, $\forall i \in \{1, \dots, n\}$, for t_{max} being the maximum flight time of the UAV, and $t(\mathbf{P}_i)$ being the time needed for following path \mathbf{P}_i .

To increase the mission safety, the final step of the pre-deployment phase verifies the paths planned for the documentation mission. First, each plan is verified by humans as collision-free by visualizing it in the 3D model of the environment. Second, the plan feasibility is verified by simulating the entire mission in the realistic Gazebo simulator using the virtual model of the environment with the same software and sensory plugins used during real-world missions. The goal of this two-stage process is to verify that all the generated paths are collision-free and do not traverse potentially risky parts of the environment. The mission specification and plan validation

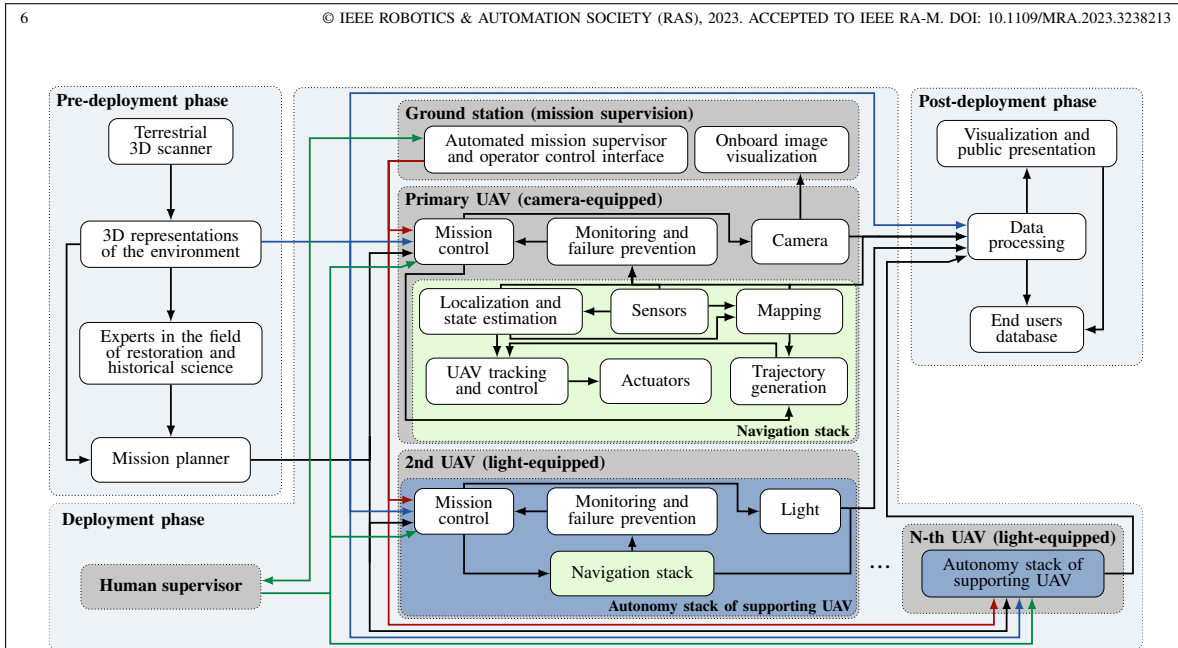


Fig. 2: High-level diagram of the three-phase architecture of the system designed for multi-UAV documentation of interiors of historical buildings. The 3D model of the environment and the mission plan are used as an a-priori generated input for the realization of the documentation mission itself. After the deployment phase, the data gathered during the mission are processed and provided to the end users.

is showcased in Fig. 3.

B. UAV Deployment

The actual system deployment is influenced by the application’s specificity imposing strict safety-guarantee requirements. After the necessary hardware checks, all software components are initialized on the onboard computer of each UAV. After successful initialization, all the UAVs automatically align their reference frame with the common frame of coordination by matching their sensory data to the sparse interior map available to each UAV. The outputs of this phase are visually verified by the operator, who checks the correctness of the frames’ alignment and validates the mission plan for the last time.

During the following autonomous mission, an automatic centralized supervisor (a ground station) checks the state of all the UAVs in real-time. This supervisor reacts to faults and allows for revealing many possible failures, even preventatively. Available safety actions include stopping all the airborne UAVs at the place at once, navigating them cooperatively to takeoff locations, and landing them at safe locations. Apart from the automatic supervisor, all these actions can be triggered by a human operator supervising the mission in parallel using the ground station. At last, a human operator serves as the final safety measure capable of landing the UAVs manually. The autonomy stack is described in section VI.

C. Post-deployment Phase

To increase the quality and range of the outputs, the data collected during autonomous flights in historical buildings are processed before being provided to the end users. This includes

post-processing of onboard sensory data to increase accuracy of pose referencing associated with the captured data frames, stitching images into photomaps, or building a 3D model of the environment in areas occluded in ground-located scans. The generated data then serve for digitalization and archivation, pre- and post-restoration analyses, state assessment and monitoring, material analyses, photogrammetry, and for digital presentation to the public.

VI. FULLY AUTONOMOUS, COOPERATING UAVS

To benefit from extensively tested and field-verified methods, the proposed multi-UAV system is based on the open source MRS UAV system² developed within the authors’ research group. In this section, let us summarize novel scientific results achieved within the presented project Dronument, whilst the MRS UAV system is described in detail in [16].

A. Reference Frame Alignment

The reference frames of the robots are aligned once during a pre-takeoff phase with each robot performing the alignment independently in four automated phases. This alignment process is mandatory for each robot as the supervising controller does not allow any robot to takeoff unless all robot frames are aligned with the global coordination frame (i.e., the map).

In the *data loading* phase, each robot loads the global map \mathbf{M} and a single 3D LiDAR data-frame \mathbf{D} to its memory, applies voxelization to both the objects for dimensionality reduction, and removes outliers in \mathbf{D} using radius outlier filter. The z-axis of both the point clouds \mathbf{M} and \mathbf{D} is assumed to be

²github.com/ctu-mrs/mrs_uav_system



Fig. 3: Pre-deployment phase of the proposed framework — (a) specification of the documentation task by selecting a set of camera viewpoints within the 3D model of the environment, (b) planning trajectory of the robot (in red) which visits all the specified viewpoints (in green), and (c) verification of the mission plan in Gazebo simulator employing identical software that is used during real-world missions.

approximately parallel to the gravity vector. During *global correlation* phase, the origins and orientations of \mathbf{M} and \mathbf{D} are approximately matched. First, convex 3D-space hulls \mathbf{H}_M and \mathbf{H}_D are computed using Qhull [17] with a hull being represented as a set of undirected edges $\mathbf{H} = \{(\mathbf{v}_a, \mathbf{v}_b)_i\}$ (set of vertex pairs). Translation $\mathbf{t}_D^M \in \mathbb{R}^3$ of \mathbf{D} to \mathbf{M} is given as $\mathbf{t}_D^M = \mathbf{b}_M - \mathbf{b}_D$, where $\mathbf{b}_X \in \mathbb{R}^3$, $\mathbf{X} \in \{\mathbf{M}, \mathbf{D}\}$, represents a polyline barycenter of an edge set \mathbf{X} as

$$\mathbf{b}_X = \frac{\sum_{(\mathbf{v}_a, \mathbf{v}_b) \in \mathbf{X}} [\mathbf{v}_a + (\mathbf{v}_b - \mathbf{v}_a)/2] \|\mathbf{v}_b - \mathbf{v}_a\|_2}{\sum_{(\mathbf{v}_a, \mathbf{v}_b) \in \mathbf{X}} \|\mathbf{v}_b - \mathbf{v}_a\|_2}. \quad (1)$$

The UAV is assumed to be taking off from ground locations, hence the grounds are coupled by setting z-axis translation to $\mathbf{t}_D^M(z) = \min_{\mathbf{p} \in \mathbf{M}} \mathbf{p}(z) - \min_{\mathbf{p} \in \mathbf{D}} \mathbf{p}(z)$, where $\mathbf{p}(z)$ denotes the z coordinate of point \mathbf{p} . Initial transformation to the consequent optimization phases is then given as

$$\mathbf{T}_1 = \mathbf{T}(\mathbf{t}_D^M) \mathbf{T}(\mathbf{t}_D, \mathbf{z}, \theta), \quad (2)$$

where $\mathbf{T} \in \mathbb{R}^{4 \times 4}$ is a general 3D transformation in the matrix form and $\mathbf{T}(\mathbf{t}_D, \mathbf{z}, \theta)$ is the matrix form of a z-axis rotation at a point $\mathbf{t}_D \in \mathbb{R}^3$ (the origin of \mathbf{D}) by angle θ . The rotation angle is given as $\theta = \theta_M - \theta_D$, where $\theta_X = \arctan \xi_y^X / \xi_x^X$, $\xi^X = (\xi_x^X, \xi_y^X, \xi_z^X) = \arg \max_{\xi \in \Xi(\mathbf{X})} \sqrt{\xi_x^2 + \xi_y^2}$, and $\Xi(\mathbf{X})$ is the set of covariance matrix eigenvectors of the point cloud \mathbf{X} .

The following *global registration* phase copes with the lateral symmetry of the environments as typical of large historical structures. Several Iterative Closest Point (ICP) routines $\text{ICP}(\mathbf{T})$ are performed in this phase, each with different initializations \mathbf{T} and loosely set parameters for point association and convergence requirements. Given a number of desired initializations k , this phase selects $\theta^* = \arg \min_{\theta \in \Theta} \text{ICP}(\mathbf{T}_1 \mathbf{T}(\mathbf{t}_D, \mathbf{z}, \theta))$ where $\Theta = \{2\pi i/k \mid i \in \{0, 1, \dots, k-1\}\}$. Final *fine-tuning optimization* phase estimates robot origin in the global coordinate frame \mathbf{T}_D^M by running $\text{ICP}(\mathbf{T}_1 \mathbf{T}(\mathbf{t}_D, \mathbf{z}, \theta^*))$ optimization set with high-accuracy parameters and strict convergence criteria.

B. State Estimation, Localization, and Mapping

Estimating the 3D state of a UAV (i.e., pose and its derivatives) in real-time is crucial for the UAV mid-air control and 3D navigation. To keep the robot steady while airborne, follow reference trajectories, and avoid obstacles, the environment needs to be perceived with robot's onboard sensors (e.g., cameras, LiDARs). As state estimation, localization, and mapping are critical for collision-free flight, the utilized algorithms are based on well-tested works with implementation validated in differing real-world scenarios. To estimate the robot state, a bank of Kalman filters [16] extended with smoothing over a short past-measurements buffer fuses onboard inertial measurements with localization outputs, providing real-time feedback to the position control loop [16]. The localization and mapping systems utilize low-drift pose estimation LOAM [18]. An extensive evaluation in [15] showed that fusing LOAM efficiently with [16] provides sufficient accuracy and robustness even in safety-critical applications. The architecture of the control, state estimation, and localization pipelines is analogous to [15]. In contrast to [15], the mapping pipeline uses an a-priori map of the environment to derive a global frame for the robots' missions (see its calibration in subsection VI-A). The a-priori shared map enables multi-robot coordination and global mission planning, but also provides an additional safety level by allowing robust online analysis of localization drift and cross-checking of sensory measurements.

C. Navigation and Trajectory Tracking

The navigation of the UAVs during the mission follows a mission plan $\mathbf{P} \in \mathbb{P}$ generated in the pre-deployment phase, described in subsection V-A. This collision-free plan is represented by a sequence of triplets $\mathbf{P} = [(\mathbf{p}_{uav}, \mathbf{p}_{ooi}, \mathbb{I})_1, \dots, (\mathbf{p}_{uav}, \mathbf{p}_{ooi}, \mathbb{I})_{|\mathbf{P}|}]$, where $\mathbb{I} \in \{0, 1\}$ is the acquisition flag. The triplets with $\mathbb{I} = 1$ specify the UAV poses \mathbf{p}_{uav} in which capturing an image or illuminating the

OoI at pose \mathbf{p}_{ooi} is required. The reference trajectory \mathbf{R} is generated by uniform sampling of the collision-free path given as sequence of $\mathbf{p}_{uav} \in \mathbf{P}$ such that the sampling step respects the required velocity. The UAV is requested to stop at each pose $\mathbf{p}_{uav} \in \mathbf{P}$ where $\mathbb{I} = 1$ to improve the quality of data acquisition by minimizing deviation from the desired pose and reducing the motion blur that would occur in case of non-zero velocity during image capturing. The reference trajectory \mathbf{R} then serves as an input to the trajectory tracking module using model predictive control (MPC). This module, described in our previous works [11], [19], produces a smooth collision-free trajectory while penalizing deviations from the original reference trajectory and respecting dynamic constraints of the UAV. The smooth-sampled reference trajectory is then passed into a feedback controller (implemented within the MRS UAV system [16]) handling tracking of the trajectory.

D. Multi-robot Coordination and Cooperation

Since the characteristics of the expected environment enable reliable use of standard communication channels, the cooperation algorithms rely on the information shared through a Wi-Fi interface among the UAVs and a ground station. Namely, the UAVs share their current poses, planned trajectories, and individual statuses based on the information from their onboard sensors. The same communication channel is utilized for commanding the UAVs from the ground station in case of emergency or a change in the mission plan, and for sharing specific messages among the UAVs during the realization of cooperative documentation techniques. The algorithms handling the autonomous flight are computed on board the UAVs.

During the cooperation, the reference trajectories of the UAVs are generated in a distributed manner on a short horizon corresponding to the optimization horizon used in the MPC-based trajectory tracking module [16]. By applying concepts of leader-follower architectures, the reference trajectories of supporting UAVs are generated with respect to the optimized trajectory of the primary UAV (leader), to the position and the desired distance of the UAV from the OoI, and to the desired lighting angle with respect to the optical axis of the documentation sensor on board the primary UAV. The coordination of the UAVs is part of trajectory optimization (see subsection VI-C) where both the current poses of the UAVs and their planned trajectories are considered to be part of constrained unfeasible space [19]. To prevent the downwash effect, this optimization is also constrained to not allow two nearby UAVs to fly above each other.

VII. AERIAL PLATFORMS

Two custom-made UAV platforms were designed specifically for the proposed application of deployment in interiors of buildings. Both the platforms, as shown in Fig. 4 and described in more detail in [20], support fully autonomous deployment within the tackled domain by carrying sensors for local environment perception together with a powerful computational unit handling the entire autonomous aerial mission. The primary platform is a heavy-weight (5.5 kg without payload) octo-rotor with dimensions of $78 \times 81 \times 40$ cm, capable

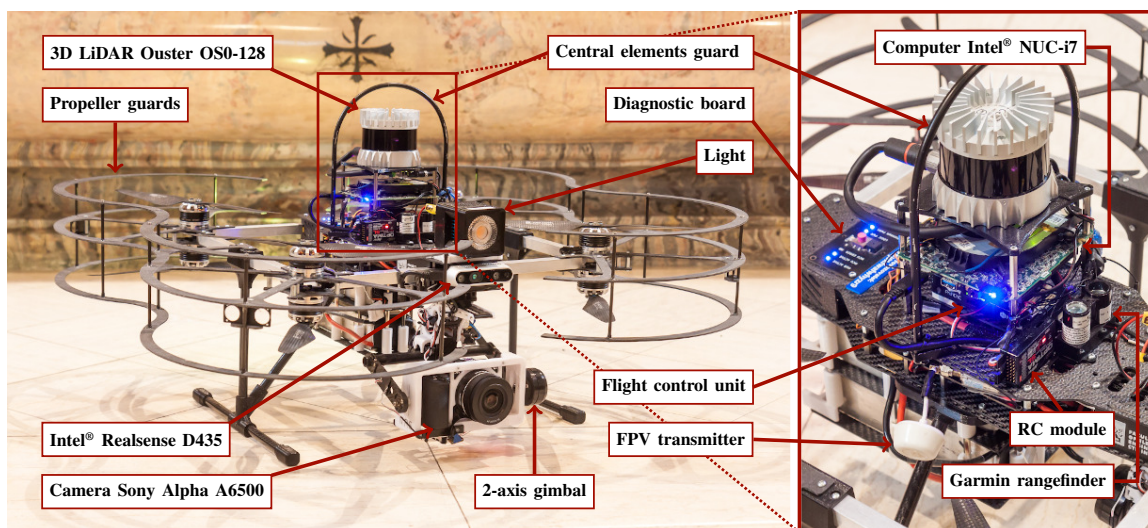
of carrying up to 1.5 kg payload — enough for a mirrorless interchangeable-lens (MIL) camera with a suitable lens and 2-axis gimbal stabilization, as well as an onboard light source. This platform minimizes its dimensions while maximizing the payload capacities, is equipped with mechanical propeller guards, and carries sensory redundancy for active obstacle avoidance. The secondary platform is a lightweight (3 kg fully loaded) quad-rotor with dimensions of $68 \times 68 \times 30$ cm suited for assisting the primary UAV throughout a documentation process by providing the scene illumination, thus increasing the quality of the gathered digital materials. While cooperating, the supporting UAVs assist in performing tasks inexecutable by a single UAV in principle. As the primary payload, the secondary platform carries a set of high-power light sources. Both the platforms support flights in close proximity to obstacles and to other UAVs. However, relative distances are limited to a minimum of 2 m to limit the aerodynamic influence of downwash, ceiling, and ground effects on the UAV, and the contrary effect of the UAV on the environment (possible damage of not firmly attached objects and fragile plasters).

A. Sensors for Autonomy

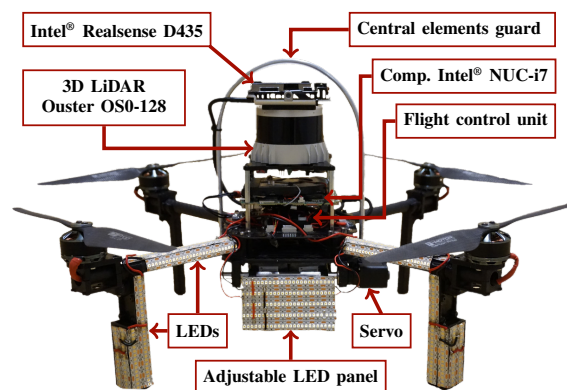
For autonomy in GNSS-denied environments, both platforms rely on onboard sensors only. The primary sensor is a 3D light detection and ranging (LiDAR) Ouster OS0-128 with 50 m detection range and 90° vertical FOV supported by a thermally-stabilized triple-redundancy inertial measurement unit, downward and upward looking point-distance sensors Garmin LiDAR Lite, and front-facing (primary UAV) or downward and upward-facing (secondary UAV) color-depth cameras Intel® Realsense D435 for sensory cross-checking in active obstacle avoidance. All the sensory data are processed by an Intel® NUC-i7 onboard computer which utilizes data in real-time algorithms handling the autonomous aerial mission. The low-level control (attitude stabilization) is handled by Pixhawk 2.1, an open-source autopilot used frequently by the robotic community. For safety reasons, the primary UAV carries a visible diagnostic RGB LED which indicates a possible failure to an operator who is authorized to override UAV autonomy for manual landing.

B. Payload

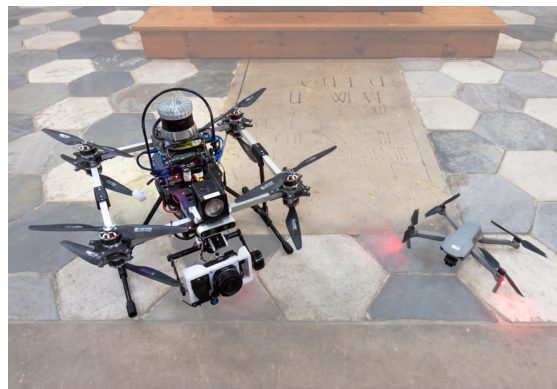
The payload equipment mountable on board the platforms is modular — cameras, lenses, and light sources can be easily interchanged for the purposes of a specific task. For general purposes, the primary UAV carries a 2-axis gimbal FlyDrotec capable of stabilizing up to 850 g payload. The stabilized axes are controllable, a feature useful mainly for controlling the pitch angle of a camera. Throughout our experiments, a MIL camera, the Sony Alpha A6500 with varying lenses, has been used for its integrated image-sensor stabilization, further minimizing the negative effect of mid-flight vibrations on the output image quality. Triggering image capture is automated via the onboard computer, whereas real-time imaging is transmitted to the ground for online visualization for the operator.



(a) Primary custom-made UAV application-tailored for documentation and inspection tasks in building interiors. The platform carries onboard sensors required for autonomous flight with equipment for acquiring high-quality documentation data, as well as a processing unit for handling autonomous flight, reasoning over the sensory data, obstacle avoidance, and the documentation mission.



(b) Secondary UAV tailored for supporting documentation tasks in building interiors. In contrast to the primary UAV (a), this platform is smaller and carries a high-power light instead of sensors for the documentation task.



(c) Comparison of the custom-made UAV platform (a) with lightweight commercial drone DJI Mavic Air 2, which carries a small camera sensor and does not support complex mission planning in building interiors.

Fig. 4: Aerial platforms used for documentation tasks in the Dronument project — primary UAV carrying documentation sensors (a), secondary UAV assisting in cooperative documentation (b), and commercial drone used for qualitative comparison (c). Both (a) and (b) carry environment-perception sensors and computational resources allowing fully autonomous deployment in interiors with poor lighting conditions.

VIII. EXPERIMENTS AND RESULTS

The extreme requirements on safety imposed by the nature of the application requiring the deployment of UAVs in priceless historical buildings imply thorough validation of all the developed software and hardware solutions prior to their deployment in real-world missions. The software solutions ranging from the state estimation and control algorithms to high-level mission control were intensively tested with the use of Gazebo simulator and the MRS simulation package³ providing realistic behavior of the UAVs. Running the same software with identical parametrization in simulation and on real hard-

³github.com/ctu-mrs/simulation

ware significantly simplifies the transfer of algorithms from the virtual environment to real-world applications. The 3D models built from the obtained 3D scans are directly used as the simulation environments for algorithms' testing. Together with simulated sensory noises and model inaccuracies, this makes the simulation as analogous to real-world conditions as possible. This methodology proves to be especially useful for discovering possible failures correlated with specific environments and validating the entire autonomous missions in an approximate copy of the real-world scenarios. Although the simulator is highly realistic, running the system in the real world introduces additional constraints. Therefore, even after thorough testing in virtual environments, the first deployments of the system

TABLE III: Overview of three-phase UAV deployment in historical buildings within the Dronument project. The first phase focused on specifying the use cases, developing the methodology, designing the system, and performing preliminary experiments, including manually controlled flights. The second phase investigated autonomous multi-robot coordination in cooperative documentation and experimented with imaging outside the visible spectrum and with the physical interaction of UAVs with the environment. The third phase deployed the system in a full-operation mode for gathering data valuable to end users and for validating the methodology and overall performance of the autonomy.

Object (approximate floor area over which the system operated)	flights	images	flight time (h:mm:ss)	flight dist. (m)	maximum height (m)	min. obs. dist. (m)	multi-robot	applied methods	
Archbishop's Chateau in Kroměříž (UNESCO, 420 m ²)	10	202	0:24:30	320	9.0	2.0	✗	VIS	Phase 1 2017–2019
Vranov nad Dyjí State Chateau (410 m ²)	18	1049	1:26:44	1020	12.0	3.0	✗	VIS	
Klein Family Mausoleum in Sobotín (30 m ²)	7	274	0:17:37	120	3.8	1.6	✗	VIS	
Rondel at State Chateau and Castle Jindřichův Hradec (140 m ²)	8	660	0:51:32	940	7.8	2.1	✗	VIS	
Chapel of All Saints at Chateau Telč (UNESCO, 84 m ²)	6	190	0:14:04	145	4.8	1.7	✗	VIS	
Church of St. Mary Magdalene in Chlumín (224 m ²)	8	86	0:23:40	146	4.8	1.5	✓	RTI	Phase 2 2019–2021
Church of the Holy Trinity in Běhařovice (252 m ²)	6	56	0:07:10	30	5.2	2.2	✗	IRF, UVF, IRR	
Church of St. Maurice in Olomouc (1160 m ²)	27	971	1:31:38	1340	16.8	1.5	✓	VIS, TPL, RAK	
Church of St. Anne and St. Jacob the Great in Stará Voda (505 m ²)	95	6022	4:54:10	4540	19.5	1.7	✓	VIS, TPL, RTI RAK, UVR, IRR	Phase 3 2021–2022
Church of the Exaltation of the Holy Cross in Prostějov (570 m ²)	7	548	0:19:49	308	15.2	1.7	✗	VIS	
Church of Our Lady of the Snows in Olomouc (918 m ²)	3	255	0:10:06	185	17.1	2.4	✗	VIS	
Church of the Assumption of the Virgin Mary in Cholín (409 m ²)	3	82	0:08:51	132	7.5	1.4	✗	VIS	
Church of the Nativity of the Virgin Mary in Nový Malín (282 m ²)	2	129	0:06:06	68	8.8	1.8	✗	VIS	
Church of the Holy Trinity in Koprivná (367 m ²)	4	211	0:17:50	247	11.4	2.0	✗	VIS	
Church of St. Bartholomew in Zábřeh (616 m ²)	4	263	0:18:23	258	12.4	1.4	✗	VIS	
Total (6387 m²)	208	10998	11:32:10	9799	19.5	1.4	✓	VIS, TPL, RTI RAK, UVR, IRR UVF, IRF	

were preceded by test flights in mock-up scenarios and testing interiors in order to reveal potential problems related to transfer of the system from simulation to real hardware.

The final version of the system, as presented in this manuscript, builds on preliminary versions and architectures of both software and hardware stacks and integrates experience from over a year and a half period of experimental deployments. During the experimental campaigns, remaining sources of potential failures were identified and the UAV system upgraded to reach the desired performance and reliability while increasing the number of realizable documentation techniques. The entire system was, to this day, deployed in real-world missions in fifteen historical buildings of various characteristics (summarized in Table III), including one of the largest Baroque halls in the Czech Republic at State Chateau Vranov nad Dyjí and the UNESCO World Heritage Sites, Archbishop's Chateau in Kroměříž and Chateau Telč. Almost twelve airborne hours in more than two hundred flights have been performed for purposes of documentation missions in the given structures. Such

an extensive experimental campaign provides an exhaustive validation of the system in real-world conditions and supports its applicability in GNSS-denied environments by identifying and overcoming challenges imposed by specific scenarios. The following sections describe the documentation techniques realized by the system in these structures. The OoIs of the presented documentation missions are showcased in Fig. 1.

A. Visible Spectrum Photography

Imaging in the visible spectrum is the most frequently applied technique as it includes methods providing the widest range of practical information while being relatively easy to perform. Within the fifteen historical structures, OoIs of various characteristics have been imaged by autonomous UAVs. These OoIs range from artistic elements, such as paintings, stained-glass windows, mosaics, stuccoes, and murals located in the most upper parts of the main naves, to complex 3D structures, such as window frames and altars up to 20 m high.

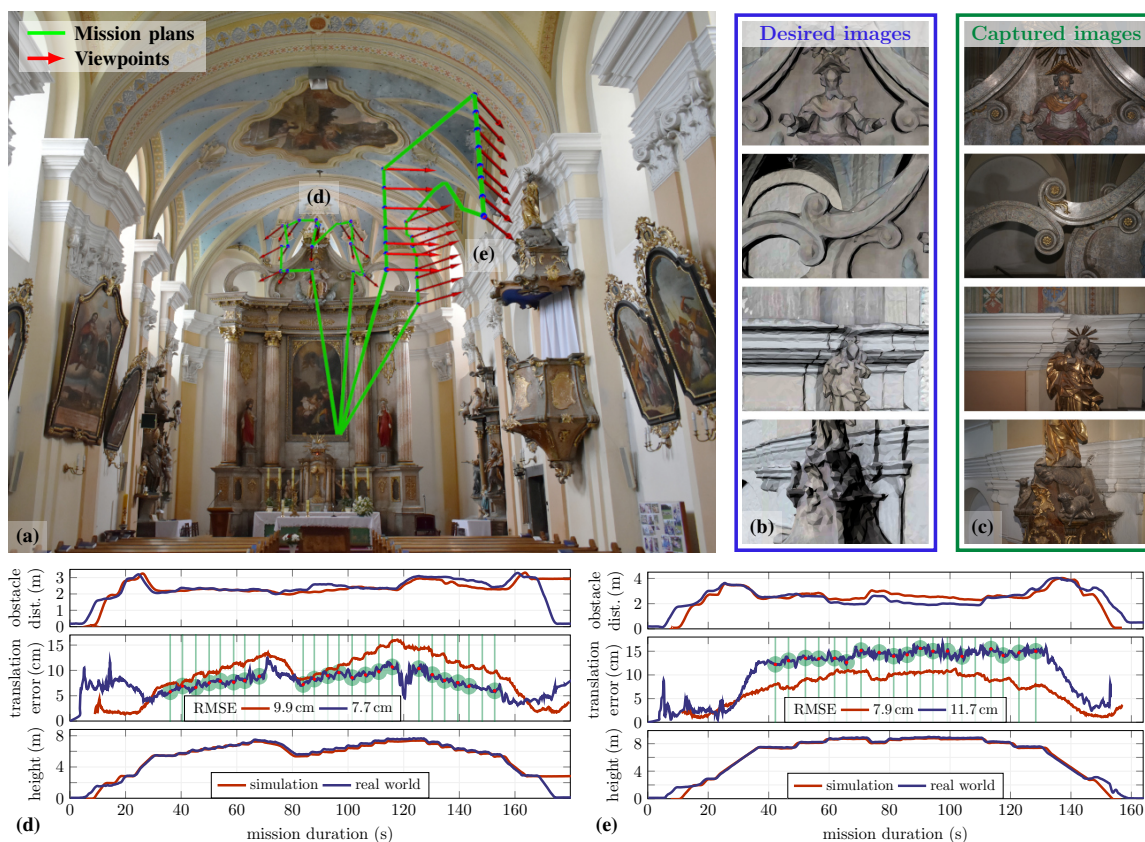


Fig. 5: An example documentation mission in the Church of the Nativity of the Virgin Mary in Nový Malín. The documentation mission was divided into two separate flights (a) focused on documenting the upper part of the altar (d) and baldachin of the pulpit (e). When going from the top, the rows in (d) and (e) show the minimal distance from the UAV frame to an obstacle, the 3D position error with image acquisition times (green vertical lines and red dots), and the height above ground. The desired imagery specified in the 3D model and the images captured on board are compared in (b) and (c). The simulation data are averaged from 5 runs each.

Additionally, objects may include structural damage, such as crevices, cracks, or fractures.

An example of fully-autonomous documentation of a single interior is provided in Fig. 5 depicting the documentation of a baroque church. The specified viewpoints were focused on documentation of two OoIs — the upper part of the altar reaching a height of 10 m and the baldachin of the pulpit. The automated process of viewpoints' specification and autonomous navigation has enabled fast realization of the documentation process in just two single-UAV flights lasting only 366 s in total. With mission specification being part of the pre-deployment phase, the overall time required for in-site deployment reached only 80 min, including equipment unpacking, flight test, mission validation and execution, and packing. Such a high level of autonomy in the process demonstrates superiority via fast, safe, effective, and repeatable data capturing when compared to the slow, imprecise, and dangerous manual control of the UAV in obstacle-filled environments by even a highly trained human operator. Even with assistive systems (stabilization and collision prevention) guiding the human in

navigation, manual operation is unsafe in losses of line of sight in the presence of obstacles and inefficient in time and accuracy required to reach the desired viewpoints. Apart from higher efficiency and safety of autonomy in contrast to human-controlled flying, a fully autonomous system allows flight in close proximity to obstacles, enlarging the operational space of the UAV. This is advantageous particularly when documenting elevated OoIs where the inaccuracy in estimating the UAV's distance to the ceiling is proportional to the distance from the human eye, thus making manual navigation in these areas unsafe.

The VIS method can be performed with commercially available products (e.g., DJI Mavic) offering semi-autonomous solutions in small and lightweight packages. However, the limited level of autonomy and sensory modularity makes the realization of the missions in large interiors prolonged (the proposed system is on average ten times faster in the same task), non-repeatable, or even impossible in conditions unfavorable to onboard perception or the desired documentation technique. In Fig. 6, the images obtained by the proposed



Fig. 6: Image outputs of VIS methodology as taken by the onboard MIL camera Sony Alpha A6500 (a) and commercial solution DJI Mavic Air 2 (b). Direct comparison of details of the images in the middle row shows that the proposed solution is superior in capturing high-quality details. This highlights the last column in which a hole in the painting is visible in top and absent in bottom image. Although the commercial solution is small and lightweight, its small sensor size of 6.4×4.8 mm hinders usability in interior documentation.

system are qualitatively compared to the ones obtained with a commercial product DJI Mavic Air 2. The figure highlights the superior performance of MIL camera imaging allowing for capturing high-resolution details of the OoIs while maintaining a safer distance from the obstacles.

Although VIS realized by a single UAV is a powerful technique, a multi-robot approach is often unavoidable if the lighting conditions are insufficient or documentation of an OoI requires non-direct lighting. An example OoI requiring additional lighting is the mural of St. Christopher in the late Gothic Church of St. Maurice in Olomouc, the documentation of which is shown in Fig. 7. Insufficient external lighting on the mural did not allow capturing bright, high-quality images without the motion blur effect arising from deviations in the reference pose over a long exposure time. Thus, to improve the quality of the images, a secondary UAV provides side lighting (approximately 45° with respect to the camera optical axis), lowering exposure times and highlighting details on the mural, such as small crevices invisible to the human eye from the ground. In the same church, 23 stained-glass windows (each about $8\text{--}34\text{ m}^2$ large) were able to be documented with a single UAV as the windows were well illuminated by the outdoor light and could be captured with short exposure times without additional lighting. The individual images of the mural and the stained-glass windows were rectified and stitched together to compose singular high-resolution orthophotos of each object. The orthophotos were used to assess the state of the OoIs for

subsequent restoration works and for enhancing the texture of the 3D model of the church⁴. As compared well in [13], the aerial-based orthophotos outperform the ground-based orthophotos in terms of quality of detail, quality of rectification due to perpendicular optical angles, and absence of occlusions.

B. Reflectance Transformation Imaging

RTI method requires a static camera and a dynamic light with a known history of poses. To validate whether the proposed system is feasible for RTI, it was applied to document a vault located 11 m above ground in St. Anne and St. Jacob the Great Church in Stará Voda (see Fig. 1b). This OoI was specifically selected as it can be photographed from a balcony on the opposite side of the central nave, thus allowing for the realization of the RTI technique in two comparable configurations: 1) with the camera (with telephoto lens) mounted on a static tripod with a clear, but misaligned view on the vault and 2) with the camera mounted on board the primary UAV. In both configurations, the light was carried on board the secondary UAV, with the directions of illumination being derived from the poses of this UAV, as estimated on board during the flight.

The comparison of results obtained in each configuration is presented in Fig. 8. The image representation produced from images captured by the tripod-mounted camera yields

⁴Selected OoIs and mapping and 3D reconstruction examples of documented historical structures can be found at mrs.felk.cvut.cz/3d-model-viewer.

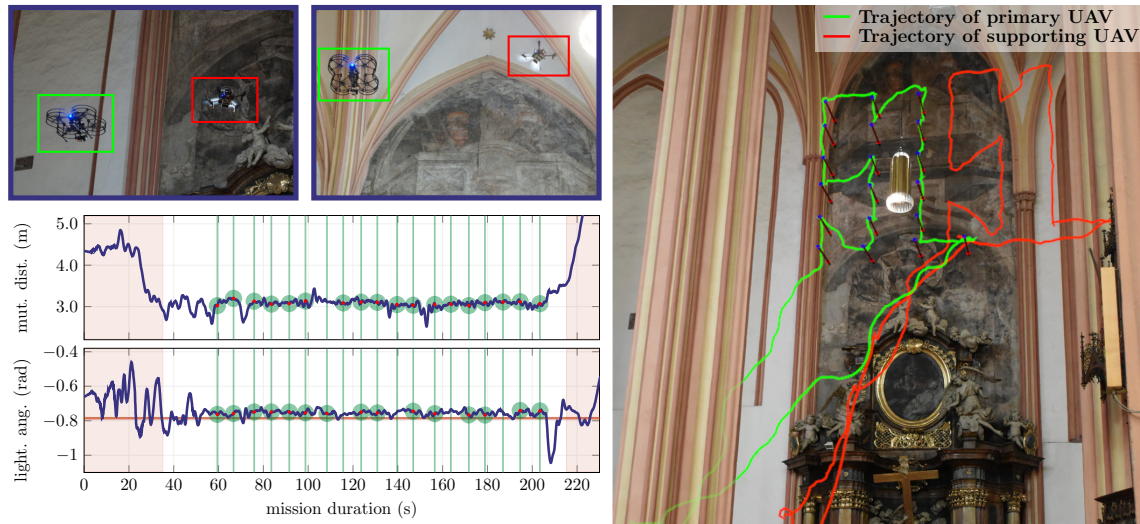


Fig. 7: Deployment of a multi-robot formation for detailed documentation of the late Gothic mural of St. Christopher using additional lighting for enhancing the quality of gathered data. The graphs show mutual distance of the UAVs during the cooperative flight and the angle between the camera optical axis and light, together with the time occasions of image capturing (green lines and circles). The red horizontal line denotes the required angle of lighting. The red areas mark parts of the mission in which the UAVs are not required to maintain the formation.

higher quality, as the choice of the OoI and usage of a telephoto lens fully compensate for the main disadvantages of the methodology in this particular case. These disadvantages are primarily smaller operational space, lower detail resolution of the resulting image caused by the large distance of the camera from the OoI, and often unavoidable occlusions. Although the fully UAV-based approach yields lower image quality since the camera's pose is not static over time, it has wider operational space and enables imaging from appropriate angles, as was verified for other OoIs in the church that could not be reasonably captured by a static camera at all. The non-staticity of the camera's reference pose misaligns the images; thus, their sub-pixel post-alignment is required to avoid blur in the resulting PTM. The experiment shows that the fully UAV-based approach yields comparable results to the single-UAV approach, which is favorable when the OoI can be photographed from the ground — an impossible scenario for most OoIs in difficult-to-reach areas of historical buildings.

C. Raking Light and Environmental Monitoring

A common feature of raking light documentation and monitoring of environmental conditions with UAVs stands in the need for robot-environment interaction. In the former, a light is attached to the wall illuminating a planar OoI from a direction perpendicular to the optical axis of the camera. This method is known to highlight even the smallest crevices and cracks in the planar surface. For the latter, a wireless sensor (e.g., for measuring humidity or temperature) is attached to the wall to measure the environmental conditions over longer periods of time. For the purpose of physical environment-UAV interaction itself, we researched a UAV equipped with a system for admittance-based control allowing for stabilization while

being attached to a planar surface (and possibly interacting with it) [12]. Before using this technology, the involved risks must be compared to the payoff, particularly inside historical buildings. To minimize the risks, it is more convenient to interact with structural (not artistic) parts of the buildings. The system was successfully tested in real-world mock-up scenarios (see Fig. 9c) with walls of sufficiently good condition.

D. IR and UV Photography

Realization of UVF and IRF (fluorescent photography) is methodically similar to VIS with the equipment being a standard MIL camera and a source of light at appropriate frequency. In contrast to VIS, the light emitted by the object illuminated by an IR or UV light source in the visible spectrum is lower. Thus, these methods require higher exposure times, as specified in Table II. The higher exposure times put stricter requirements on image stabilization in the presence of onboard vibrations, inaccuracies, and disturbances that cause UAVs to deviate from their reference pose.

Realization of the UV and IR reflectography requires a camera without UV and IR filters and exposure times of tens of seconds. This makes the use of UAVs for imaging in UV and IR reflectography unfeasible. However, supporting ground-based imaging with aerial lighting is applicable. The UAVs can carry (relatively close to the OoI) high-power LEDs radiating in the desired spectrum. The IR and UV-based methods were tested in St. Anne and St. Jacob the Great Church, Stará Voda (see Fig. 9) and in Church of the Holy Trinity, Běhařovice. The experiments showed that the proposed system can be used in realization of the UV and IR-based methods in historical structures, even in limited lighting conditions.

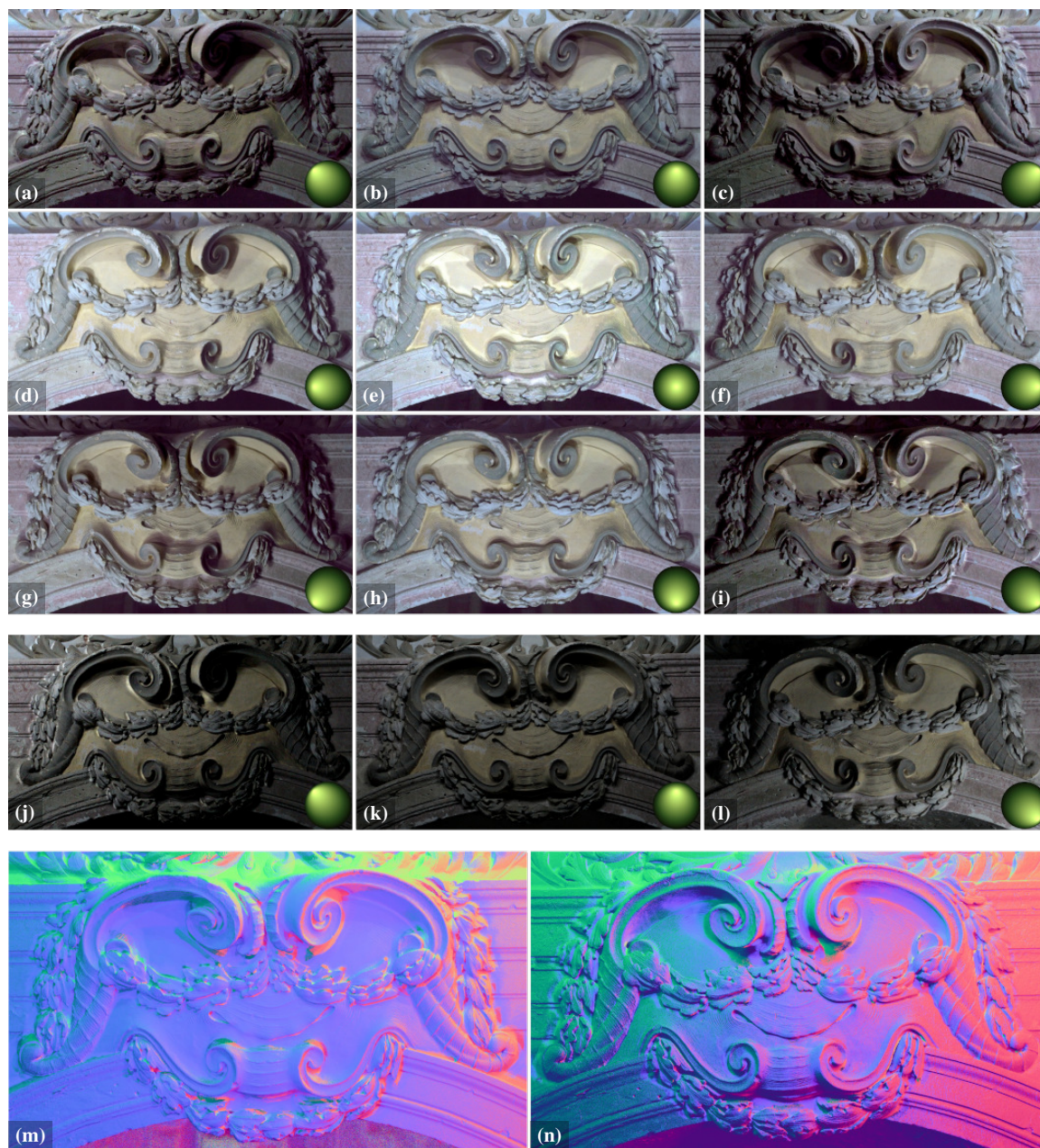


Fig. 8: Comparison of polynomial texture maps (PTM) obtained with a fully UAV-based RTI approach with camera carried by a UAV (a)–(i) and PTM obtained from images taken by a camera mounted on a static tripod (j)–(l). In both cases, the dynamic positioning of light is provided by the secondary UAV. The bottom row shows the normal maps encoded in RGB for fully UAV-based approach (m) and a single UAV approach (n).

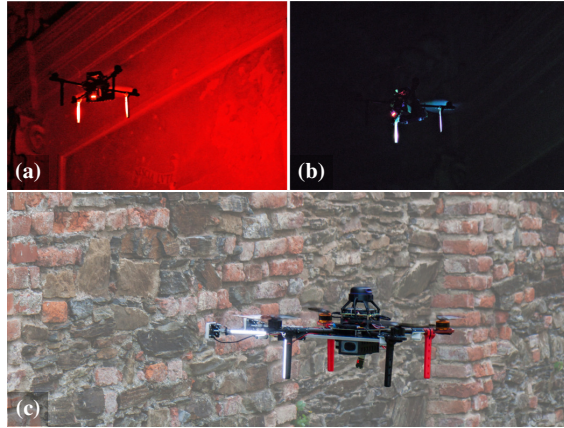


Fig. 9: Deployment of UAVs carrying IR (a) and UV (b) source of light, and a frame-extension mechanism for physical attachment and interaction with static planar surfaces (c).

E. Mapping and 3D Reconstruction

The capacities of UAVs allow capturing the interior under difficult-to-reach angles, not only for imaging purposes, but also for spatial mapping of the structures. Although terrestrial laser scanners yield the most accurate maps, these devices cannot, in principle, document occluded spaces, whereas the larger operational space of UAVs allows for minimizing these occlusions. This advantage is showcased in Fig. 10 where above-edge areas could not be reconstructed from scans captured from ground. The potential for accurate 3D mapping using UAVs is immense; however, is not the main purpose of the proposed system which outputs dense 3D maps only as a byproduct to the photo-documentation task. The onboard-UAV-built maps contain larger amounts of noise as the mobile laser-scanning technology is less accurate (lightweight, low-power, and moving while scanning) than static scanners, making it harder to align the captured scans, even in post-processing. To achieve the best results for 3D reconstruction, we recommend leveraging the advantages of both methodologies simultaneously.

IX. DISCUSSION

The proposed UAV-based system for documenting historical monuments of differing structures, dimensions, and complexity has demonstrated its wide applicability in real-world documentation tasks, ranging from RGB photography and 3D mapping to multi-robot RTI in areas high above the ground. The high level of autonomy, the ability to fly beyond the visual line of sight between the UAV and a human operator, and the deployability in low lighting conditions (using a worldwide unique method of dynamic illumination by a cooperating UAV team) enable to gather crucial data for heritage protection and documentation that was not possible before. This universally novel system has been used in the very first fully-autonomous multi-robot real-world deployments in such complex and safety-demanding interior structures.

However, deploying mobile robots inherently poses risks to the environment, humans, and equipment therein. This requires

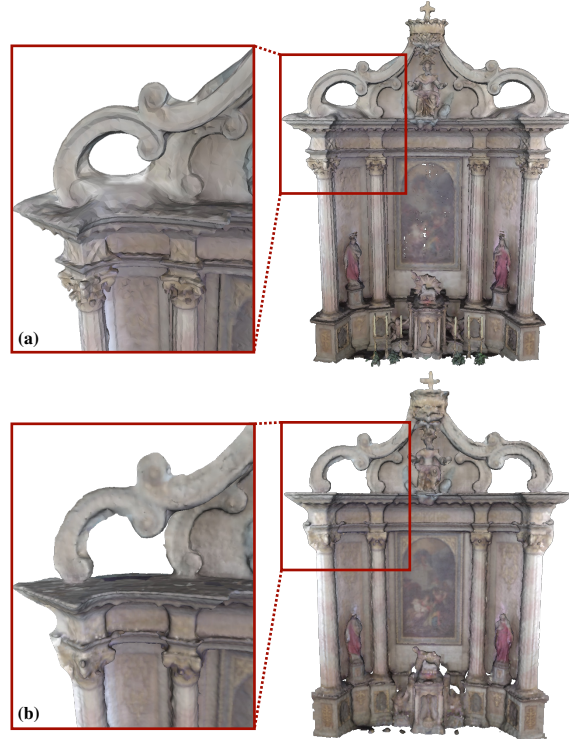


Fig. 10: 3D reconstruction of the altar at the Church of the Nativity of the Virgin Mary in Nový Malín, Czech Republic. The altar reconstructions were done using scans obtained by (a) terrestrial laser scanner Leica BLK360 and (b) Ouster OS0-128 mounted on board an autonomous UAV during the deployment shown in Fig. 5. The meshes were created with the Poisson surface reconstruction and colored using the panoramic RGB images captured by the terrestrial scanner.

careful justification of the UAVs' use that, in our experience, tends to be needlessly overused — conventional technology provides a safer and better quality solution in many documentation tasks. A common example is imaging the interior ceiling or low-height OoIs, where using a static camera with a long-focus lens was identified to be a more appropriate solution. Manual-control UAV solutions are also sufficient if the task is small-scale, the lighting conditions are feasible, repeatability is not required, and the OoIs are few. The need for multi-UAV teams in tasks achievable with sufficient quality by a single UAV, such as the selected example of single-UAV RTI presented in Fig. 8, should also be considered prior a full-scale deployment.

X. CONCLUSION

This work has presented a universally novel study on an autonomous multi-robot UAV-based system for realization of advanced documentation techniques in culturally valuable environments. The system showcases the immense potential of mobile robots for fast, accurate, and mobile digitalization of difficult-to-access interiors. The hardware and software architectures of the self-contained autonomous-UAV-based system

were introduced and experimentally validated through almost twelve hours of flight time in more than two hundred real-world flights of single-UAVs and multi-UAV teams in fifteen historical monuments of varying structures. The system design has emerged from close cooperation with a team of restorers, and the data collected during the autonomous missions has been used by the end users in successive restoration works.

The study also assists in identifying the current challenges and future directions of research in aerial documentation and inspection. Based on the high added value for heritage protection, the system has been approved by the Czech National Heritage Institute for indoor usage and is accompanied by an official methodology (available at [1]) describing the proper usage of UAVs in historical structures. It is the first methodology of this authority for using UAVs in historical buildings and so prescribes the system to be a standard in this application.

XI. ACKNOWLEDGMENT

This work was supported by the Ministry of Culture of the Czech Republic through project no. DG18P02OVV069 in program NAKI II, by the Ministry of Education of the Czech Republic through OP VVV funded project CZ.02.1.01/0.0/0.0/16 019/0000765 "Research Center for Informatics," by the European Union's Horizon 2020 research and innovation program AERIAL-CORE under grant agreement no. 871479, by CTU grant no. SGS20/174/OHK3/3T/13, and by the Czech Science Foundation (GAČR) under research project No. 20-10280S.

The authors would like to thank Nicolas Staub for an initial survey of the projects' applicability in documentation and restoration, Jan Bednar for his assistance in creating 3D models for public presentation, Pavel Stoudek for preparing hardware platforms used during the experimental analyses, and Vojtech Krajicek for technical consultation of the restoration techniques. At last, we would like to thank to representatives of the Czech National Heritage Institute, namely Milan Skobrtal and Michaela Cadilova, for our fruitful cooperation during the project and for the opportunity to deploy the proposed system in real-world structures.

REFERENCES

- [1] Multi-Robot Systems group, CTU FEE. (2022) Dronument (Dron & Dronument). [Online]. Available: <http://mrs.felk.cvut.cz/dronument>
- [2] S. Prieto, B. Quintana, A. Adan, and A. Vazquez, "As-is building-structure reconstruction from a probabilistic next best scan approach," *Robotics and Autonomous Systems*, vol. 94, pp. 186–207, 2017.
- [3] P. S. Blaer and P. K. Allen, "Data acquisition and view planning for 3-D modeling tasks," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2007, pp. 417–422.
- [4] D. Borrmann, R. Heß, H. R. Houshiar, D. Eck, K. Schilling, and A. Nüchter, "Robotic mapping of cultural heritage sites," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XL-5/W4, pp. 9–16, 2015.
- [5] T. Bakirman, B. Bayram, B. Akpınar, M. F. Karabulut, O. C. Bayrak, A. Yigitoglu, and D. Z. Seker, "Implementation of ultra-light uav systems for cultural heritage documentation," *Journal of Cultural Heritage*, vol. 44, pp. 174–184, 2020.
- [6] T. Özasan, G. Loiano, J. Keller, C. J. Taylor, V. Kumar, J. M. Wozen-craft, and T. Hood, "Autonomous navigation and mapping for inspection of penstocks and tunnels with mavs," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1740–1747, 2017.
- [7] M. Beul, D. Droschel, M. Nieuwenhuisen, J. Quenzel, S. Houben, and S. Behnke, "Fast autonomous flight in warehouses for inventory applications," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3121–3128, 2018.
- [8] M. Ceccarelli, D. Cafolla, G. Carbone, M. Russo, M. Cigola, L. J. Senatore, A. Gallozzi, R. Di Maccio, F. Ferrante, F. Bolici, S. Supino, N. Colella, M. Bianchi, C. Intrisano, G. Recinto, A. Micheli, D. Vistocco, M. R. Nuccio, and M. Porcelli, "HeritageBot Service Robot assisting in Cultural Heritage," in *IEEE International Conference on Robotic Computing*, 2017, pp. 440–445.
- [9] N. Hallermann, G. Morgenthal, and V. Rodehorst, "Vision-based monitoring of heritage monuments — Unmanned Aerial Systems (UAS) for detailed inspection and high-accurate survey of structures," *WIT Transactions on The Built Environment*, vol. 153, pp. 621–632, 2015.
- [10] P. Petracek, V. Kratky, and M. Saska, "Dronument: System for Reliable Deployment of Micro Aerial Vehicles in Dark Areas of Large Historical Monuments," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2078–2085, 2020.
- [11] V. Kratky, P. Petracek, V. Spurny, and M. Saska, "Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2302–2309, 2020.
- [12] D. Smrcka, T. Baca, T. Nascimento, and M. Saska, "Admittance Force-Based UAV-Wall Stabilization and Press Exertion for Documentation and Inspection of Historical Buildings," in *International Conference on Unmanned Aircraft Systems*, 2021, pp. 552–559.
- [13] V. Kratky, P. Petracek, T. Nascimento, M. Cadilova, M. Skobrtal, P. Stoudek, and M. Saska, "Safe Documentation of Historical Monuments by an Autonomous Unmanned Aerial Vehicle," *ISPRS International Journal of Geo-Information*, vol. 10, no. 11, pp. 7381–16, 2021.
- [14] S. Lin and B. W. Kernighan, "An effective heuristic algorithm for the traveling-salesman problem," *Operations Research*, vol. 21, no. 2, pp. 498–516, 1973.
- [15] V. Kratky, P. Petracek, T. Baca, and M. Saska, "An autonomous unmanned aerial vehicle system for fast exploration of large complex indoor environments," *Journal of Field Robotics*, vol. 38, no. 8, pp. 1036–1058, 2021.
- [16] T. Baca, M. Petrlik, M. Vrba, V. Spurny, R. Penicka, D. Hert, and M. Saska, "The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles," *Journal of Intelligent & Robotic Systems*, vol. 102, no. 26, pp. 1–28, 2021.
- [17] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The Quickhull Algorithm for Convex Hulls," *ACM Trans. Math. Softw.*, vol. 22, no. 4, pp. 469–483, 1996.
- [18] J. Zhang and S. Singh, "Low-drift and real-time lidar odometry and mapping," *Autonomous Robots*, vol. 41, no. 2, pp. 401–416, 2017.
- [19] M. Saska, V. Kratky, V. Spurny, and T. Baca, "Documentation of dark areas of large historical buildings by a formation of unmanned aerial vehicles using model predictive control," in *IEEE International Conference on Emerging Technologies and Factory Automation*, 2017, pp. 1–8.
- [20] D. Hert, T. Baca, P. Petracek, V. Kratky, V. Spurny, M. Petrlik, M. Vrba, D. Zaitlik, P. Stoudek, V. Walter, P. Stepan, J. Horyna, V. Pritzl, G. Silano, D. Bonilla Licea, P. Stibinger, R. Penicka, T. Nascimento, and M. Saska, "MRS Modular UAV Hardware Platforms for Supporting Research in Real-World Outdoor and Indoor Environments," in *IEEE International Conference on Unmanned Aircraft Systems*, 2022, pp. 1264–1273.

- [4c] M. Petrlik, **P. Petracek**, V. Kratky, T. Musil, Y. Stasinchuk, M. Vrba, T. Baca, D. Hert, M. Pecka, T. Svoboda, and M. Saska, “UAVs Beneath the Surface: Cooperative Autonomy for Subterranean Search and Rescue in DARPA SubT,” *Field Robotics*, vol. 3, no. 1, pp. 1–68, 2023

Our fruitful research endeavors in the autonomous exploration of subterranean environments began in [3c, 8a, 12a, 133] and resulted in another core work of this thesis [4c], which was published in *Field Robotics*. The work is motivated by the Subterranean Challenge robotic competition¹, which was organized to test and push the limits of state-of-the-art field robotics by developing robotic solutions for assisting first responders in their dangerous work. The work proposes a state-of-the-art, self-contained, and heavily field-tested stack for autonomous and distributed stabilization, control, perception, multi-sensor mapping, and communication of a team of autonomous robots with the main objective of cooperative navigation and exploration in harsh, challenging, and unknown subterranean environments. Within the scope of the task, the proposed solution is capable of fast 3D exploration in vast, unknown, dynamic, and complex environments containing large open spaces as well as narrow passages. The stack utilizes small-factor UAVs capable of fast 3D navigation with distributed sensors and computational power in perceptually degraded and constrained environments, which are a priori unknown.

Given our premise defined in Chapter 1 and the extreme constraints of the DARPA SubT, emphasis is put on verifying the algorithms’ robustness concerning the challenging real-world conditions and real-time performance of all of the tightly-coupled algorithms running on-board resource-constrained UAVs (shown in Figure 3.3). The tight coupling of the entire stack enables fast reactions to changes in the environment, thereby enabling the effective use of the limited flight time of UAVs. The proposed autonomy stack allows for seamless cooperation in efficient exploration between aerial and ground vehicles if intermittent robot-to-robot communication is provided. The contributions include achievements in solving the full autonomy in a heterogeneous workspace and the transition between varying environments of human-made tunnels [133], urban buildings [8a], and cave systems [3c].

Among others, the author’s contributions presented in [4c] include a novel method for preventing LiDAR-perception degeneracy of a UAV flying in whirling dust clouds, which are a common source of perceptual degradation in a plethora of subterranean environments, as well as in urban and historical buildings with low humidity. Due to wind gusts and the aerodynamic effects of the UAV, the dust particles circulate and noise the neighborhood around the UAV. These particles degrade the perception of the UAV and may lead, apart from map degeneration, to crashes as a result of unfeasible or noisy state estimation. The contributions of [4c] include filtering of dust particles from a 3D point cloud. The method builds on the fact that the energy of a light beam reflected back to the LiDAR sensor is only partial when a dust particle is hit. Given this, the reflected light beams with the lowest energy are most likely to be the measurements of noise. As the aerodynamic influence of the UAV is limited locally and energy from distant measurements is assumed to dissipate in the environment, a local threshold-based filtration is applied on the intensity field of the sensor’s raw data. This method has proved reliable even in the harshest dust conditions of subterranean environments, where cameras were completely blinded by whirling dust. When using the proposed method, the processed LiDAR perceived the environments without degradation and measurement noise.

¹The author was a member of team CTU-CRAS-NORLAB comprised of members from Czech Technical University and Université Laval, Canada.

As part of the contributions, the author of this thesis has introduced and integrated methods of 3D perception (LiDAR data processing, mapping, SLAM) to a state-of-the-art open-source² system for UAV control — the MRS UAV System [39]. The pipelines for these methods were prepared for deploying teams of UAVs in various indoor and outdoor environments given an arbitrary number of onboard sensors measuring spatial information. This modular methodology was and still is in use for various research and industrial projects and robotic competitions within the MRS laboratory, as well as by international users of the open-source system. These contributions include adaptation, parallelization, and optimization of the state-of-the-art LiDAR odometry LOAM [42] for its use in real-time on-board UAVs and coupling it with inertial modality and probabilistic volumetric mapping for generating dense metric maps. This dense mapping enabled further research of path and trajectory planning and complex-environment navigation and exploration in [4c, 8a, 23, 134]. The contributions to this method lie in open-sourcing its optimized mapping-coupled version, as well as its thorough qualitative and quantitative analysis in [4c] and in [3c, 8a, 10a, 12a, 134].

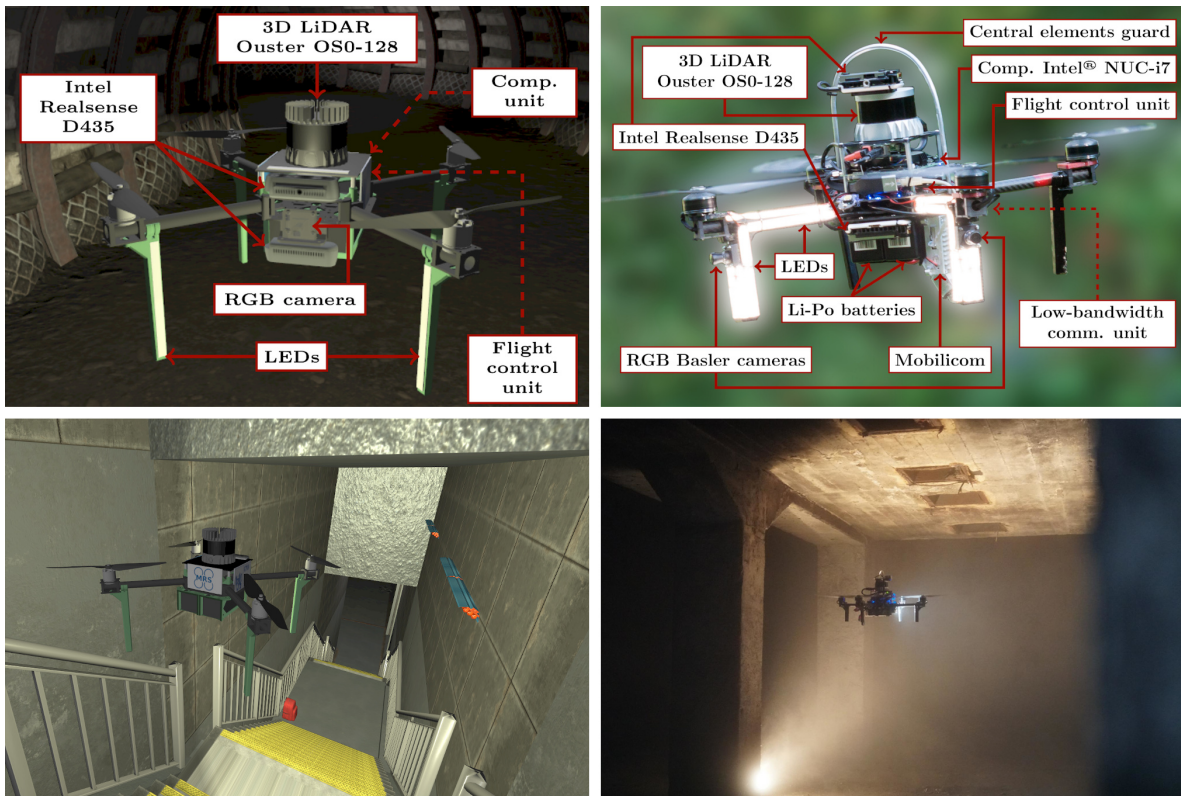


Figure 3.3: Aerial platform used by team CTU-CRAS-NORLAB in the final event of the DARPA SubT in the virtual (left column) and real-world (right column) environments.

²MRS UAV System is openly available at github.com/ctu-mrs/mrs_uav_system.

Special Issue: DARPA Subterranean Challenge, Advancement and Lessons Learned from the Finals (DARPA SubT Final)

Field Report

UAVs Beneath the Surface: Cooperative Autonomy for Subterranean Search and Rescue in DARPA SubT

Matěj Petrлік¹, Pavel Petráček¹, Vít Krátký¹, Tomáš Musil¹, Yurii Stasinchuk¹, Matouš Vrba¹, Tomáš Báča¹, Daniel Heřt¹, Martin Pecka¹, Tomáš Svoboda¹ and Martin Saska¹

Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic

Abstract: This paper presents a novel approach for autonomous cooperating UAVs in search and rescue operations in subterranean domains with complex topology. The proposed system was ranked second in the Virtual Track of the DARPA SubT Finals as part of the team CTU-CRAS-NORLAB. In contrast to the winning solution that was developed specifically for the Virtual Track, the proposed solution also proved to be a robust system for deployment onboard physical UAVs flying in the extremely harsh and confined environment of the real-world competition. The proposed approach enables fully autonomous and decentralized deployment of a UAV team with seamless simulation-to-world transfer, and proves its advantage over less mobile UGV teams in the flyable space of diverse environments. The main contributions of the paper are present in the mapping and navigation pipelines. The mapping approach employs novel map representations—SphereMap for efficient risk-aware long-distance planning, FacetMap for surface coverage, and the compressed topological-volumetric LTVMap for allowing multirobot cooperation under low-bandwidth communication. These representations are used in navigation together with novel methods for visibility-constrained informed search in a general 3D environment with no assumptions about the environment structure, while balancing deep exploration with sensor-coverage exploitation. The proposed solution also includes a visual-perception pipeline for on-board detection and localization of objects of interest in four RGB stream at 5 Hz each without a dedicated GPU. Apart from participation in the DARPA SubT, the performance of the UAV system is supported by extensive experimental verification in diverse environments with both qualitative and quantitative evaluation.

Keywords: Unmanned Aerial Vehicles, Search and Rescue, DARPA, SubT, autonomy, exploration, navigation, deployment, subterranean environment, degraded sensing

Support materials

The paper is supported by the multimedia materials available at mrs.felk.cvut.cz/fr2022darpa. Open-source implementation of the core of the UAV system is available at github.com/ctu-mrs/

Received: 6 June 2022; revised: 7 October 2022; accepted: 29 November 2022; published: 2 January 2023.

Correspondence: Matěj Petrлік, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic, Email: matej.petrlik@fel.cvut.cz

This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Copyright © 2023 Petrлік, Petráček, Krátký, Musil, Stasinchuk, Vrba, Báča, Heřt, Pecka, Svoboda and Saska

DOI: <https://doi.org/10.55417/fr.2023001>

[mrs_uav_system](#). The SLAM datasets are available at github.com/ctu-mrs/slam_datasets. The visual detection datasets are available at github.com/ctu-mrs/vision_datasets.

1. Introduction

The research of new robotic technologies and solutions is accelerating at an unprecedented rate mainly in case of aerial robotics. Technological development is improving many areas of our lives and, hopefully, even the future of humanity. The authors of (Shakhatreh et al., 2019) reviewed current research trends and future insights on potential Unmanned Aerial Vehicle (UAV) use for reducing risks and costs in civil infrastructure. The survey of UAV applications is accompanied by a discussion of arising research challenges and possible ways to approach them.

This paper focuses on a robotic system developed to autonomously search subterranean environments. The motivation behind searching subterranean environments is to gain situational awareness and assist specialized personnel in specific missions. Such missions may include: assessing the structural integrity of collapsed buildings, tunnels, or mines; exploration of a newly discovered branch in a cave network; or searching for lost persons. These tasks can often be life-threatening to human workers as many hazards are present in subterranean environments. In order to reach survivors quickly in unstable caves or partially collapsed burning buildings, first responders, such as emergency rescuers and firefighters, may potentially put their lives at risk. In firefighting tasks, fires can be either localized and reported to personnel by robots or the robots can even directly begin extinguishing flames if the presence of human firefighters is too risky (Spurny et al., 2021; Pritzl et al., 2021; Martinez-Rozas et al., 2022). In such scenarios, ceilings can suddenly collapse, toxic gas can appear in a mine, flames can extend to an escape corridor, or a cave cavity can flood with water. In distress situations, it is essential to swiftly coordinate the rescue operation as the survivors of a catastrophe might need acute medical assistance or have a limited amount of resources available, namely oxygen and water. However, without conducting a proper reconnaissance of the environment and assessing the potential risks prior to the rescue mission, the involved rescuers are exposed to a much higher probability of injury.

To reduce the possibility of bodily harm or to avoid risks altogether, a robotic system can be sent on-site before the rescuers in order to either quickly scout the environment and report any hazards detected by the onboard sensors, or directly search for the survivors. The rescue mission can be further sped up by deploying a team of robots capable of covering larger areas and offer redundancy in case of losses of some robot units in harsh environments. Multirobot teams can also consist of heterogeneous agents with unique locomotion modalities to ensure traversability of various terrains, including muddy ground, stairs, and windows, which is discussed in the overview of collaborative Search and Rescue (S&R) systems (Queralta et al., 2020). Similarly, sensing modalities can be distributed among individual robots to detect various signs of hazards, such as increased methane levels or the potential presence of survivors deduced from visual or audio cues. Mounting all sensors on a single platform would negatively affect its dimensions and, consequently, its terrain traversability as it may not be able to fit into narrow passages, such as crawlspace-sized tunnels or doorways. It would also mean a single point of failure for the rescue operation. On the other hand, the operation of a single robot can be managed by just one person, while commanding a robot team may be unfeasible for a single operator. Assigning each robot to an individual operator would also be an ineffective allocation of resources. Moreover, the range of the robot would be limited by the communication link to the operator. To provide a valuable tool for the rescue team, the robots must be able to move through the environment on their own and infer about the environment using their sensor data. The rescuer can then also act as an operator, providing only high-level inputs to the robotic system to bias their behavior based on *a priori* information (e.g., someone was last seen on the east side of the third floor). The research and development of such autonomous systems for assisting first responders is the primary focus of the S&R robotics, and also the motivation for the S&R UAV system presented in this paper.

The robotic platforms typically considered for S&R tasks are categorized into wheeled, tracked, legged, marine, and aerial platforms (Delmerico et al., 2019). Among these locomotive modalities,

aerial robots are considered to have the highest traversal capabilities since they can fly over most obstacles which are untraversable by other platforms. One example of an autonomous aerial research platform for S&R is found in (Tomic et al., 2012). The mobility of UAVs also surpasses other robot types thanks to its dynamic flight which can achieve large velocities and accelerations. These qualities make UAVs ideal for swift environmental scouting for gaining initial knowledge about a situation. As such, the aerial platform is predetermined to be deployed as the first robot during the first minutes of the rescue operation. A team deployed in an outdoor multi-UAV disaster response task (Alotaibi et al., 2019) can effectively cover a large search area and minimize the time to find and reach survivors. On the other hand, UAVs cannot operate for extended periods of time due to their limited flight time, and the sensory equipment is limited by the maximum payload of the UAV. Some sensing modalities might even be unsuitable for the use on aerial robots due to their propulsion system, e.g., detecting gas due to the aerodynamic effects of the propellers, or sound detection due to noisy operation. Due to the aforementioned pros and cons of UAV platforms, it is convenient to combine the capabilities of other robot types to form a heterogeneous robotic team.

This manuscript proposes an autonomous cooperative UAV approach for S&R. The approach used by Unmanned Ground Vehicles (UGVs) is not presented here because it is vastly different from the UAV system and as such would not fit into the scope of this article, which is already moderately extensive as we did not want to omit any details about the deployed system. The UGV solution was developed by our colleagues who are acknowledged at the end of this article. The proposed UAV together with legged, wheeled, and tracked UGVs formed the CTU-CRAS-NORLAB team, which participated in the Defense Advanced Research Projects Agency (DARPA) Subterranean Challenge (SubT). The team consisted of Czech Technical University in Prague (CTU) and Laval University.

1.1. DARPA SubT challenge

After major success in accelerating the development of self-driving cars in the Grand Challenges of 2004 and 2005 and the Urban Challenge in 2007, DARPA announced the Subterranean Challenge (SubT) (Orekhov and Chung, 2022) for the years 2017-2021 to advance the state of the art of S&R robotics. Participants had to develop robotic solutions for searching subterranean environments for specific objects that would yield points if reported with sufficient accuracy. To achieve the task at hand, the competitors had to develop complex multirobot systems spanning nearly all research areas of mobile robotics, from design of the robotic platforms to high-level mission planning and decision-making.

The rules of the competition can be summarized in a few points. Each team has a dedicated time slot, or *run*, to send their robots into a previously unvisited course and search for specific objects, referred to as artifacts (Figure 1). Each run starts at a predefined time and ends exactly one hour later. A single team is present on the course at a time during which they can deploy an unconstrained number of robots of arbitrary size. The movement of team personnel and their handling of robots is allowed only in the area in front of the entrance to the course, as shown in Figure 2. Only robots can enter the course and only one human operator/supervisor can command the robots and access



Figure 1. All 10 artifacts searched for in the Final Event of DARPA SubT (image courtesy of DARPA). The operator had to submit the position of the identified artifact with accuracy better than 5 m. While the first three artifacts (survivor, cellphone, and backpack) were present in all circuits, the drill and the fire extinguisher were tunnel-specific. Similarly, the gas and vent were located in the urban environment, and the helmet with rope could be found in the caves. The last artifact (the cube) was introduced only for the Final Event.

4 · Petrlík et al.

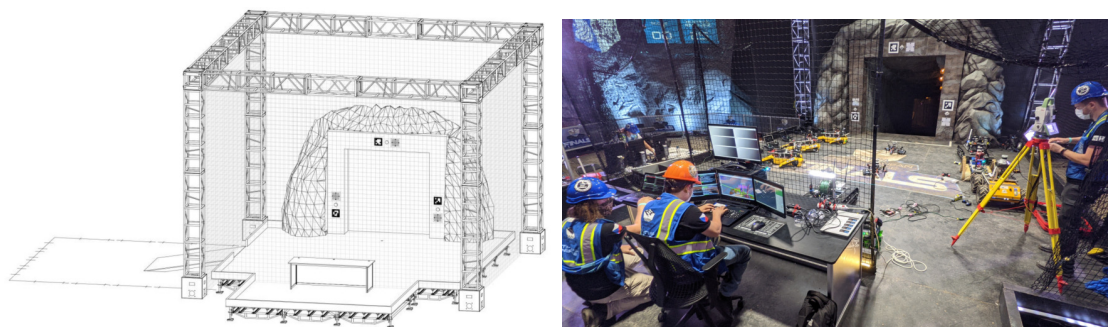


Figure 2. The bounded staging area (image courtesy of DARPA) is the only place where the human crew members can handle the robots. The person sitting behind the displays is the operator who is the only one allowed to issue commands to the team of robots, and also to view and interpret mission data.

Table 1. The prize money awarded for achieving the first three places in the Final Event.

Place	Systems Track	Virtual Track
1.	\$2M	\$750K
2.	\$1M	\$500K
3.	\$500K	\$250K

the data they acquire during the run. These conditions should mimic the conditions of a real S&R robotic mission. The operator can report the type and position of an artifact. If the type was correct and the reported position was not further than 5 m from the true position, the team was awarded one point. The team with the highest score wins the prize according to [Table 1](#). For a more detailed description of the challenge, see ([Orehov and Chung, 2022](#)).

To encourage the development of high-level components without worrying about the resilience of the hardware in harsh subterranean conditions and also to enable teams without many resources and/or physical robots to compete, a virtual version (Virtual Track) of the competition was run in parallel to the physical Systems Track. The solutions of the Virtual Track were uploaded as Docker images (one image per robot) to the Gazebo-based Cloudsim simulation environment, where the entire run was simulated. Every team could use the Cloudsim simulator to test their approaches in practice worlds prior to the actual competition.

The competition was further subdivided into individual circuits, which were events in the specific subterranean environments of a tunnel, cave, and urban space. Examples of each environment are shown in [Figure 3](#). The surroundings were chosen to correlate with typical real S&R sites to assure the applicability of the systems developed during the competition. Every type of environment differs in size, geometric dimensions, traversability conditions, and requirements on perception modalities. The specifics of tunnel-like environments are summarized in ([Tardioli et al., 2019](#)) with 10 years of experience in S&R ground robots research. The role of mobile robots in rescue missions after mine disasters is discussed in ([Murphy et al., 2009](#)). The Final Event combined all of the previous environments for the ultimate challenge.

We participated in the competition first as a nonsponsored team. In the Tunnel Circuit, we won 1st place among the nonsponsored teams and 3rd place in total, which earned us \$200 000. The aerial robots explored 340 m of tunnels and found 3 artifacts out of the 10 artifacts discovered by all our robots ([Petrlik et al., 2020](#)). This success was repeated in the Urban Circuit with the same place achieved but this with time larger prize money \$500 000. The UAVs proved their suitability for quick scouting of the environment thanks to their advantage in mobility when the first deployed UAV managed to travel 93 m inside the building in just 200 s while it took about half an hour to reach the same area with the semi autonomously operated ground robots as reported in ([Kratky et al., 2021a](#)). One of the deployed UAVs also served as a retranslating station for other robots

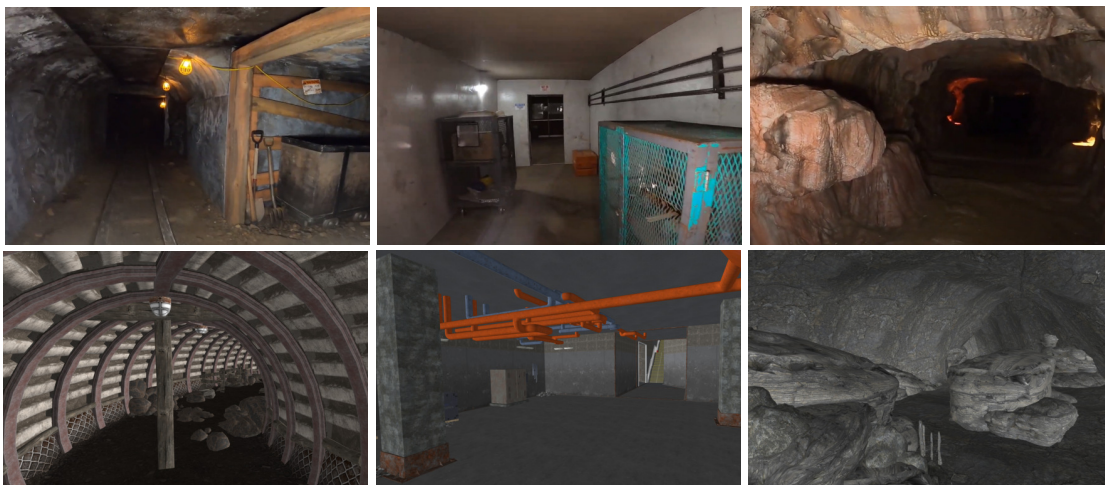


Figure 3. Three types of subterranean environments found in the competition, each challenging for the robot team in a different way. From left to right: tunnel, urban, and cave. The top row shows examples of environments from the Systems Track of the Final Event, while the virtual worlds are pictured in the bottom row.

after navigating to and landing at a strategic position. Thanks to consistent performance in both circuits, DARPA awarded our team the funding for the Final Event, which allowed us to acquire more capable hardware. In the Virtual Track, the UAVs were used as the primary platform for finding artifacts thanks to their high travel speed and the ability to fly over terrain untraversable by UGVs. The ground robots supported longer flights of the UAVs by extending the communication network with breadcrumbs. In total, 215 artifacts were found by the UAVs in the competition worlds (8 artifacts less than the winner). The performance of UAVs in the confined environment of the Systems Track was worse than in the Virtual Track. Nevertheless, while the UGVs detected 5 out of 7 scored artifacts, the aerial robots managed to add 2 unique artifacts not seen by other robots but 1 of them had the wrong class and image and thus could not score. The last point was scored manually by the operator by matching a detection from a UGV that had inconsistent map with the same position in a correct map. The approach presented in this paper is the result of UAV research, development, and testing over the whole 3-year-long period.

2. Related work

The state of the art in rescue robotics is coherently summarized in the survey (Delmerico et al., 2019), which concerns both hardware and software. On the hardware side, different robot morphologies, locomotion types, and platform designs are categorized. Regarding software, the survey concerns perception and control algorithms. The authors interviewed experts on disaster response and humanitarian aid to understand the situation and needs of rescuers.

Here, we provide an overview of the solutions for perception in adverse conditions of the underground environments, methods of localization and mapping for precise and reliable navigation, and techniques for safe traversal of narrow corridors. A summary of systems deployed in previous circuits of DARPA SubT follows. Finally, relevant datasets are referenced in order to prompt further research effort in the S&R area.

2.1. Degraded sensing

Perception in subterranean environments faces constant degradation of the sensor outputs due to the harsh conditions of such places. The underground climate is often filled with impervious dust (particularly in mines), where any movement agitates the settled layer of fine dirt and mineral

particles. On the other hand, caves are typically humid ecosystems, where dense mud replaces the dust layer found in mines. However, the elevated humidity forms droplets of fog, which corrupt the measurements of most visible or Near Infrared (NIR) light-based sensor modalities, and also causes frequent reflections on wet surfaces. Radars can reliably penetrate smoke, dust, and fog, and after postprocessing using, e.g., Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), a 2D occupancy grid for navigation (Lu et al., 2020) can be constructed. Another reliable sensing modality for when images from color (RGB) cameras are polluted by dust or fog is thermal imaging, which, in (Khattak et al., 2019), is used for the localization of robots in areas with airborne obscurants. Our approach goes beyond these works by employing intensity-based filtering of the Light Detection and Ranging (LiDAR) data, and thus no additional sensors are necessary even in dense clouds of dust.

2.2. Localization and mapping

Recent developments in S&R robotics sparked the research of more precise local pose estimation algorithms (also referred to as odometry), as well as long-term globally consistent trajectory and multirobot map fusion of all agents of the robotic team. The state-of-the-art methods were published in (Cadena et al., 2016), where the challenges and future direction of the Simultaneous Localization and Mapping (SLAM) development are also identified. The demands on low control error and robustness to degraded sensor data in the narrow subterranean environments present in the DARPA SubT pushed all contesting teams to either adapt and improve an existing method to be usable in the extreme conditions, or to develop a new SLAM tailored to this specific domain. SLAM methods used by the teams in the Final Event are summarized in (Ebadi et al., 2022) along with expert opinions about the present maturity and future outlook of the field.

Team CoSTAR developed a LiDAR odometry solution (Palieri et al., 2020) based on Generalized Iterative Closest Point (GICP) matching of LiDAR scans with initialization from Inertial Measurement Unit (IMU) and wheel odometry, including the possibility of extension to other odometry sources, such as Visual-Inertial Odometry (VIO). The method is shown to outperform state-of-the-art localization methods on the datasets from Tunnel and Urban circuits. An ablation study presents the influence of individual components on the total Absolute Position Error (APE). The second improved version (Reinke et al., 2022), which was released as open-source is less computationally demanding, less memory intensive, and more robust to sensor failures. All presented experiments are conducted with ground robots. The localization of aerial vehicles is handled by a resilient HeRo state estimation system (Santamaria-Navarro et al., 2019). The state estimation stack considers heterogeneity and redundancy in both sensing and state estimation algorithms in order to ensure safe operation, even under the failure of some modules. Failures are detected by performing confidence tests on both data and algorithm health. If a check does not pass successfully, the resiliency logic switches to the algorithm with the best confidence, similar to our previous solution published in (Baca et al., 2021). The local odometry of (Palieri et al., 2020; Santamaria-Navarro et al., 2019) is accompanied by loop closure detection and pose graph optimization locally on each robot, as well as globally on the base station. This optimizes the trajectories of all robots for a multirobot centralized SLAM solution (Ebadi et al., 2020). After improving the performance of the multirobot loop closure generation and pose estimation, especially in large-scale underground environments, the solution was open-sourced (Chang et al., 2022) and released together with a multirobot dataset from the subterranean environment. A technique for loop closure prioritization (Denniston et al., 2022) improves the Absolute Trajectory Error (ATE) of the multirobot SLAM by prioritizing loop closures based on observability, graph information, and Received Signal Strength Indicator (RSSI) criteria. A decentralized SLAM solution for UAVs (Lajoie et al., 2020) performs distributed outlier-resilient pose graph optimization when another agent is within communication range. This method can be used with either a stereo camera or a LiDAR, and is evaluated on a dataset from the Tunnel Circuit.

The long, featureless corridors that are often present in man-made tunnels lead to unobservability of the motion along the degenerate direction, which leads to significant drift. Promising approaches,

such as (Shan et al., 2020; Xu et al., 2022), constrain the solution of the optimization problem using the preintegrated IMU measurements. This helps to reduce the localization drift under unfavorable environmental geometry. Nevertheless, the vibrations induced by spinning propellers degrade the inertial measurements, and can thus negatively affect the localization precision. Approaches, such as those seen in (Ebadi et al., 2021), detect the geometrical degeneracy using the ratio of the most observable and the least observable directions. This ratio is then used to determine loop closure candidates to reduce the drift along the degenerate direction. Similarly, (Zhang et al., 2016) handles environment degeneracy in state estimation by not updating the solution in detected degenerate directions. Another possibility is to combine the 3D LiDAR method with a direct visual odometry method [e.g., (Alismail et al., 2016)], which tracks image patches by minimizing the photometric error. This approach, which is shown in (Shin et al., 2020), has the advantage over feature-based methods like that of (Zhang and Singh, 2015) in that it provides low drift, even when salient image and geometric features are lacking. The disadvantage is that localization performance is worsened when whirling dust is present in the camera image, as reported in (Petrlik et al., 2020).

Team CERBERUS developed a complementary multimodal sensor fusion (Khattak et al., 2020). The odometry estimated by visual/thermal inertial odometry is used as a prior for LiDAR scan-to-scan and scan-to-map matching. The VIO/TIO priors constrain the scan matching optimization problem, thus reducing drift in a degenerate environment significantly, which is demonstrated in an experiment conducted in a self-similar environment.

Another multimodal approach is the Super Odometry (Zhao et al., 2021) of team Explorer, which was deployed on aerial robots in the tunnel and urban circuits of DARPA SubT. The core of the method is the IMU odometry with biases constrained by VIO and LiDAR-Inertial Odometry (LIO), which are initialized with preintegrated inertial measurements of the constrained IMU. The relative pose factors of VIO and LIO are weighted based on the visual and geometrical degradation, respectively.

Team MARBLE first relied on visual SLAM (Kramer et al., 2021), but after Subterranean Integration Exercise (STIX), they transitioned to the LiDAR-based Cartographer (Hess et al., 2016) due to unstable tracking of motion under poor illumination, reflections, dust, and other visual degradation.

Wildcat SLAM (Hudson et al., 2022) of the CSIRO Data61 team is a multiagent decentralized solution, where each agent computes a global map using the currently available data shared among the robots. The odometry of each agent is based on the work of (Bosse et al., 2012).

Our approach is similar to the other teams' as we also use primarily LiDAR for localization and mapping. An improvement over the state of the art is the compensation of the delay (Pritzl et al., 2022a) caused by the LiDAR scan processing and the delay of the localization itself.

2.3. Mobility

Deploying aerial robots has one great advantage over ground robots due to their full terrain traversability. A UAV can fly over terrain that would compromise the safety of an UGV, e.g., steep decline, mud, water, etc. This allows to neglect the traversability problem necessarily tackled in solutions to UGV navigation (Fan et al., 2021), as the only movement constraint of aerial platforms flying through an enclosed environment is the minimum size of a passage that the robot can safely pass through. The dimensions of such passages depend largely on the size of the UAV, but also on the precision of the pose estimation, the control error of onboard regulators, the map quality, and the reactive behavior in close vicinity of obstacles. Some platforms also tolerate contact with obstacles in the sense that the contact does not endanger the continuation of the mission (Huang et al., 2019). Other types of platforms adapt their morphology and/or locomotion modality to their current surroundings and obstacles (Fabris et al., 2021). In voxel-based map representations, the size of a narrow passage is represented too conservatively, i.e., the size of the narrow passage in the voxel map is the lower bound of the true size. However, in practice, the narrow passage can be up to twice the map resolution larger than its voxel representation, which prevents traversing passages

that are well within the physical limits of the UAV. To better approximate the true shape of the narrow passage, (O’Meadhra et al., 2018) propose continuous representation based on Gaussian Mixture Models (GMM) (Reynolds, 2009), which is converted to a voxel map of arbitrary resolution when queried. However, the information about the exact structure of the environment is lost due to the approximation by Gaussian distribution. We took another approach of locally increasing the resolution of the occupancy voxel map when the size of the environment requires it, which preserves all details.

To fully exploit the capabilities of UAV’s full terrain traversability, the path planning and trajectory generation algorithms have to work with a full 3D representation of the environment and fulfil the real-time requirements. Although several excellent works on planning in constrained environments were recently published (Zhou et al., 2021a; Tordesillas et al., 2022), they focus primarily on generating fast trajectories while the presented application requires maximizing the reliability of the system in the presence of uncertainties imposed by a harsh dynamic environment. In addition, deploying the planning algorithm as part of the complex system running on board UAV with limited computational resources motivates the use of computationally undemanding algorithms. Similarly to team Explorer (Scherer et al., 2022), we make use of a multistage approach consisting of extensively validated computationally undemanding algorithms well-integrated into presented system (Baca et al., 2021; Kratky et al., 2021a).

2.4. DARPA SubT approaches

This paper primarily focuses on the approach developed for and experimentally verified in the Final Event of DARPA SubT. As mentioned, these results are built upon the experience in using the approaches developed for the tunnel and urban circuits. The practical verification of the developed solutions in challenging environments justifies the robustness of these algorithms. Valuable insights on the future of S&R robotics can be drawn from lessons learned by the teams.

Team CoSTAR relied on their uncertainty-aware framework, NeBula, in the tunnel and urban circuits (Agha et al., 2021). The framework supports multimodal perception and localization including radar, sonar, and thermal cameras. Aerial robots were part of their heterogeneous team in STIX and the tunnel circuit, mainly for exploring areas inaccessible to ground robots and data muling with distributed data sharing (Ginting et al., 2021; Saboia et al., 2022). A reactive autonomy approach COMPRA (Lindqvist et al., 2021) was also proposed for UAV underground S&R missions. Their solution gained 2nd and 1st place in the tunnel and urban circuits respectively.

Team Explorer developed a system (Scherer et al., 2022) that achieved 1st place in the tunnel circuit and 2nd place in the urban circuit. Their collision-tolerant platform “DS” with flight time of 13min was carried on top of a UGV and could be launched by the operator when needed. The authors identified the challenge of combined exploration and coverage problem when their UAVs with limited camera Field Of View (FOV) missed some artifacts along their flight path. The frontier-based exploration pipeline used a custom OpenVDB mapping structure (Museth, 2013) for sampling frontier-clearing viewpoints. Paths to found viewpoints were planned using bidirectional RRT-Connect.

Team CERBERUS deployed legged ANYMAL robots and aerial DJI Matrice M100 robots in the tunnel circuit. Their graph-based system for the autonomous exploration of subterranean environments called GBPlanner was deployed in multiple locations. The exploration of Edgar mine during STIX and the National Institute for Occupational Safety & Health (NIOSH) mine during the tunnel circuit are documented in (Dang et al., 2020b). Specifically, the exploration method for aerial robots (Dang et al., 2019a) consists of a local fast-response layer for planning short collision-free paths and a global layer that steers the exploration towards unvisited parts of the map. This method is part of the solution for underground search by aerial robots found in (Dang et al., 2020a). A mapping and navigation approach (Papachristos et al., 2019a) for autonomous aerial robots based on the next-best-view planner (Papachristos et al., 2017; Bircher et al., 2016) was also proposed, but was later outperformed by the GBPlanner (Dang et al., 2020b). The uncertainty in localization

and mapping is taken into account during the planning in (Papachristos et al., 2019b) in such a way that among all trajectories arriving to the reference waypoint, the one that minimizes the expected localization and mapping uncertainty is selected. To unify the exploration framework across both legged and aerial platforms, (Kulkarni et al., 2021) have revised (Dang et al., 2020b) and added a cooperation framework that identifies global frontiers in a global graph built from the sub-maps of individual robots. The unified strategy for subterranean exploration using legged and aerial robots in tunnel and urban circuits is presented in (Tranzatto et al., 2022b). Team CERBERUS won in the Systems Track of the Final Event and (Tranzatto et al., 2022a) describes their approach that led to this success.

Team MARBLE presents their system deployed to STIX, the tunnel circuit, and the urban circuit in (Ohradzansky et al., 2021). The aerial robots relied on direct vision-based local reactive control and map-based global path planning. Global path planning is common with ground and aerial robots. Viewpoints are selected based on the frontier voxels covered by the camera FOV and the approximate travel time. In the tunnel circuit, the local reactive control generates velocity commands by steering the UAV towards a look-ahead point from the global path, while being repulsed by nearby obstacles. With this planner, traversing narrow passages was problematic due to noise in the depth image. Thus a new planner was developed for the urban circuit based on voxel-based probabilistic tracking of obstacles (Ahmad et al., 2021). In the Systems Track of the Final Event, team MARBLE gained 3rd place.

A heterogeneous team of robots including UAVs was also deployed by team CSIRO Data61 (Hudson et al., 2022), both in the tunnel and urban circuits. The aerial part of the team consisted of a DJI M210 equipped with the commercially available payload of Emesent Hovermap, and a custom gimballed camera. To explore the environment of the urban circuit, the autonomy utilized an approach based on the direct point cloud visibility (Williams et al., 2020). Team CSIRO Data61 achieved 2nd place in the Systems Track of the Final Event.

Although team NCTU did not participate in the Final Event, their solution (Chen-Lung et al., 2022) to the tunnel and urban circuit showcased originality in the form of autonomous visually localized blimps (Huang et al., 2019). Their navigation was based on policies learned by deep reinforcement learning with simulation-to-world transfer.

Our CTU-CRAS-NORLAB team first participated in the STIX event with a hexarotor platform localized by optic flow (Walter et al., 2018) of the downward-facing camera. The reactive navigation used LiDAR scans to stay in the middle of the tunnel and move forward in a preferred direction at an intersection. The predictive controller (Baca et al., 2016) was forgiving to imprecise localization caused by strenuous optic flow estimation in the whirling dust of the tunnels. The heterogeneous team that secured 3rd place in the tunnel circuit (Roucek et al., 2019) consisted of wheeled, tracked, and aerial robots with different sensor payloads. Instead of unreliable optic flow, the localization of the UAV system (Petrlik et al., 2020) was revamped to rely on 2D LiDAR, HectorSLAM (Kohlbrecher et al., 2011), and state estimation (Petrlik et al., 2021). The hardware platform was also downscaled to a 450 mm diameter quadrotor. The vertical element of the urban circuit called for upgrading the LiDAR to a 3D one, which consequently required a redesign of the whole navigation pipeline (Kratky et al., 2021a) to allow for six Degrees of Freedom (DOF) mobility through the 3D environment. Physically, the platform was based on the same frame as what was used in the tunnel circuit, however prop guards were added to reduce the chance of destructive collision while flying through doors. The CTU-CRAS-NORLAB approach to the urban circuit, which we completed in 3rd place, is described in (Roucek et al., 2020). Although the cave circuit was canceled, extensive preparations were still performed in the sizable Bull Rock cave in South Moravia (Petracek et al., 2021). The exploration depth of the UAV team was greatly extended by a multirobot coordinated homing strategy that focused on extending the communication range of the base station by landing the returning UAVs on the edge of the signal. Based on the lessons learned during these competition and testing deployments (during the 3 years of development UAVs of the CTU-CRAS-NORLAB team achieved > 400 flights and traveled > 50 km in demanding real world environments) the new approaches presented in this paper were designed.

2.5. Datasets

Due to the challenging nature of the subterranean environments, such as narrow passages, degenerate geometry, and perception degradation, datasets that were collected by the competing teams are valuable to the community as the algorithms can be evaluated on demanding data degraded by the previously mentioned issues. In contrast to the verification often conducted under artificially ideal lab conditions, these datasets present a fair way to compare algorithms in realistic conditions. A SLAM dataset (Rogers et al., 2020a) collected during the tunnel circuit and STIX consists of LiDAR scans, images from a stereo camera and thermal camera, IMU measurements, and RSSI, together with a professionally surveyed ground truth map and measured artifact positions. The dataset from the urban circuit (Rogers et al., 2020b) was recorded using the same sensors with the exception of an added carbon dioxide (CO₂) sensor and the lack of a thermal camera. Data from sensors used for autonomous navigation including color-depth (RGBD) camera, event camera, thermal camera, 2D and 3D LiDARs, IMU, and Ultra-Wide Band (UWB) positioning systems were collected (Koval et al., 2022) by a mobile robotic platform moving through a subterranean environment. Another dataset (Kasper et al., 2019) for comparison of VIO methods contains outdoor, indoor, tunnel, and mine sequences, with ground truth poses obtained by laser tracking the sensors rig. Aerial datasets consisting of unsynchronized LiDAR scans and IMU measurements from UAVs flying in the cave, tunnel, and mine environments are included in this paper,¹ with ground truth poses estimated using a professionally surveyed ground truth map. We also publish the labeled visual detection datasets² consisting of images from both UAV and UGV cameras that were used for training of the artifact detection Convolutional Neural Network (CNN). Images from the Tunnel and Urban circuits, Bull Rock Cave, and industrial buildings are included.

3. Contributions

An approach for cooperative exploration of demanding subterranean environments by a team of fully autonomous UAVs in S&R tasks is presented in this paper. Deployment of this approach in the DARPA SubT virtual competition was awarded by 2nd place. The simulation model of the UAV platform designed by our team was used by seven out of nine teams. The crucial contributions of the developed system can be summarized in the following list:

- **A complex approach that can serve as a guide for building a system for Global Navigation Satellite System (GNSS)-denied operations.** The proposed approach was extensively verified in numerous simulated worlds and real physical environments ranging from vast caves, industrial buildings, tunnels, and mines to large outdoor openings. Most importantly, the UAVs were deployed into the intentionally harsh conditions of the DARPA SubT to push them to their limits. The experience gained from hundreds of flights in such conditions are condensed into the lessons learned presented in this paper, which we deem valuable for the field robotics community.
- **Novel mapping structures** are proposed for safety-aware reactive planning over large distances, for compact volumetric inter-robot information sharing, for storing coverage of surfaces by onboard sensors, and for finding a suitable landing spot.
- **Maximization of the probability of detecting a nearby artifact** by searching not only the unexplored space, but also visually covering known surfaces while respecting the limited field of view of the onboard sensors. The detection is coupled with probabilistic estimation of artifact positions based on multitarget tracking and detection-to-hypothesis association, which improves the precision of artifact localization while the robot is moving around the artifact.
- **A novel safety-aware approach to planning** that considers the risk of planned trajectories in addition to the path length in the optimized cost function. In contrast to the state-of-the-art

¹github.com/ctu-mrs/slam_datasets

²github.com/ctu-mrs/vision_datasets

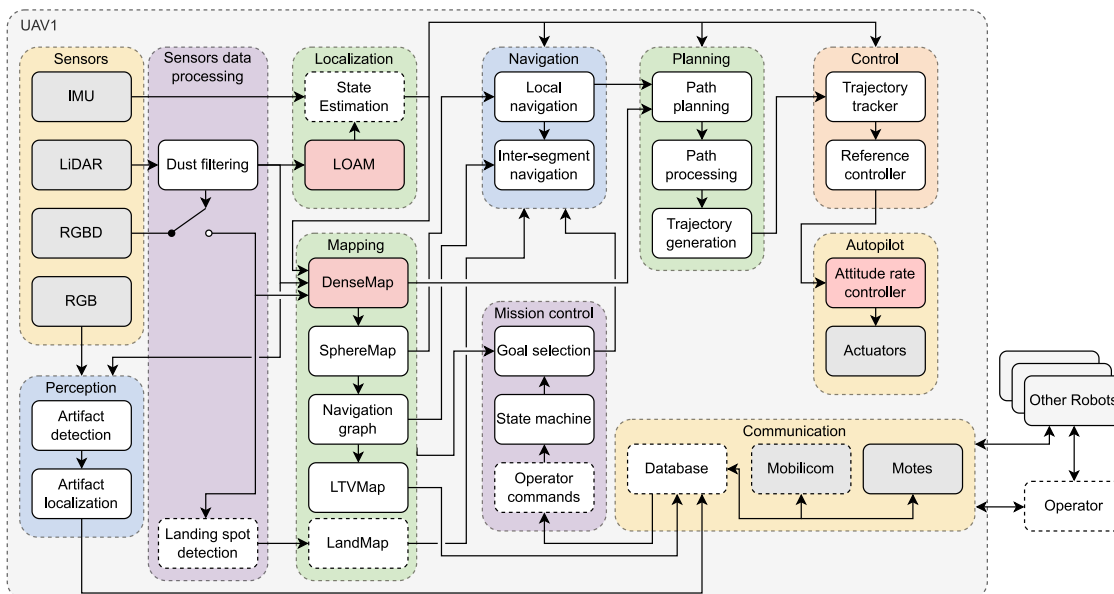


Figure 4. The diagram shows individual modules of the UAV system architecture (as deployed on the hardware platform) grouped into logical categories. Hardware modules are filled with gray, and red distinguishes open source modules not developed by us. The modules marked by dashed lines were used only in Systems Track but not in Virtual Track.

methods, longer paths are selected if the estimated risk of collision is lower than the risk of a shorter path.

- **Full autonomy of the UAV** allows for scalability of the size of the deployed fleet without placing additional workload on the operator. Nevertheless, the operator can override the autonomy with one of the available special commands to change the default search behavior when the UAV is in communication range.
- **The multirobot autonomous search** benefits from a higher number of deployed UAVs that share their topological representations of the environment to cooperatively cover a larger area by biasing the search towards parts unvisited by other agents.

4. System architecture overview

The whole autonomous system of a single UAV consists of software modules, each with different inputs, outputs, and purpose. These modules and their interconnections are depicted in Figure 4 with the individual modules grouped into more general logical categories. The first category includes the physical *Sensors* (Section 5) of the UAV—the IMU, LiDAR, RGB, and RGBD cameras. The description of the important parameters of the used sensors is available in Section 12. Measurements from IMU and LiDAR enter the *Localization* group (Section 6), where a full-state estimate of the UAV is obtained. LiDAR is also used in combination with the RGBD camera for building maps in the *Mapping* module group (Section 7). The *Perception* (Section 10) category focuses on detection and localization of artifacts using all the available sensor data.

Autonomous search through the environment is governed by the *Mission control* category (Section 11), which selects goals (Section 8) based on the current state of the state machine, models of the environment from the *Mapping* group, and possibly also commands from the operator. A coarse path consisting of waypoints to the selected goals is found by the *Navigation* (Section 9.1) and further refined and time-parametrized in the *Planning* modules (Section 9) in order to produce a safe and dynamically feasible trajectory. The *Control* blocks (Baca et al., 2021) track the

trajectory and generate attitude rate references for the low-level *Autopilot* that controls the actuators (Section 12).

The operator receives crucial mission status data, topological maps, and, most importantly, detected artifacts through the *Communication* layer (Roucek et al., 2020). This also allows the operator to influence or override the autonomous behavior of the UAV. All transmitted data are received by other UAVs (or other robots, in the case of a heterogeneous team) in the communication range, which serves two purposes: one, the receiving agent can propagate the message further down the network, and, two, the topological maps allow penalizing goals already visited by other robots to better allocate resources over a large area.

5. Spatial perception

The equipment on board UAV platforms within our research group is modular and replaceable to support a wide spectrum of research areas (Hert et al., 2022). In the proposed system for agile subterranean navigation, however, the aerial platform is fixed to ease fine-tuning of the on-board-running algorithms. From the point of perception, it relies heavily on 3D LiDAR from Ouster (SLAM, dense mapping, and artifact localization), and utilizes vertically oriented RGBD cameras for filling space out of FOV of the primary LiDAR sensor, and uses two RGB Basler cameras for artifact detection, supported by powerful LEDs illuminating the scene. The flow of sensory data within the entire system are shown directly in Figure 4.

5.1. Sensors calibration

The intrinsics of LiDAR sensor and RGBD cameras are factory-calibrated whilst monocular RGB cameras are calibrated with standard OpenCV calibration tools, assuming the pinhole camera model. The extrinsics of the sensors (cameras-to-LiDAR and LiDAR to the flight control unit) are given by the CAD model of the robot. To mitigate the effects of inaccuracies in 3D printing, modeling, and assembly, all the camera-to-LiDAR extrinsics are fine-calibrated using a checkerboard camera calibration pattern with known dimensions. The fine-calibration pipeline detects the pattern in both modalities (LiDAR data and RGB image), finds mutual correspondences, and estimates the extrinsics by defining the problem as perspective-n-point optimization minimizing the reprojection error of the mutual correspondences with Levenberg-Marquardt method.

5.2. Filtering observation noise

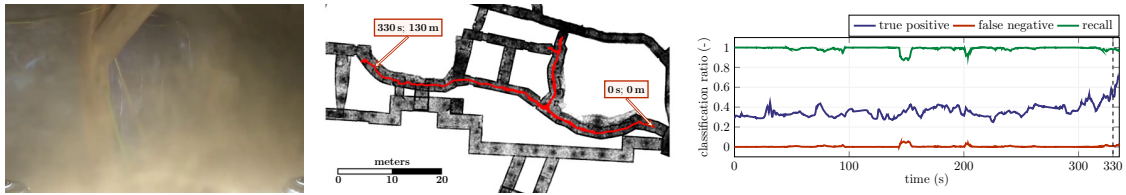
The aerodynamic influence of a multirotor UAV on the environment is not negligible, particularly in confined settings. The fast-rotating propellers generate airflow lifting up light particles of dust and whirling them up in clouds. In environments where the clouds are not blown away but are rather rebounded back to the UAV, the effect on sensory performance might be crippling. To minimize deterioration in perception and its dependent systems (e.g., mapping, localization), the incident noise is filtered out from local LiDAR data.

The idea of robust filtering of dust is based on the method presented in (Kratky et al., 2021a) in which LiDAR data are sorted by the intensity field (measured intensity of the reflected light for a given point) and 10% of the lowest-intensity data in a local radius from the sensor are removed. In contrast to the baseline method, simpler thresholding is adopted such that a subset $\mathcal{P}_F \subseteq \mathcal{P}$ of LiDAR data \mathcal{P} is preserved. The absence of data sorting lowers the computational load and reduces delay in data processing. The set is given as $\mathcal{P}_F = \mathcal{P}_D \cup \mathcal{P}_I$, where

$$\mathcal{P}_D = \{\mathbf{p} \mid \|\mathbf{p}\| \geq \kappa, \mathbf{p} \in \mathcal{P}\}, \quad (1)$$

$$\mathcal{P}_I = \{\mathbf{p} \mid \mathcal{I}(\mathbf{p}) > \Upsilon, \mathbf{p} \in \mathcal{P} \setminus \mathcal{P}_D\}. \quad (2)$$

$\mathcal{I}(\mathbf{p})$ (W m^{-2}) is the intensity of the reflected light from a point \mathbf{p} , κ (m) is a local radius of a filtering sphere with LiDAR data origin at its center, and Υ (W m^{-2}) is the minimal intensity of preserved



(a) Dense cloud dust around the UAV as viewed in onboard RGB camera at time 330 s. (b) Top view on the UAV trajectory. (c) Performance of noise classification in LiDAR data in 3 m local radius from the sensor. Average recall reached 99 %.

Figure 5. LiDAR-data noise filtration running onboard a UAV during a 154 m flight in the mine part (the dustiest part) of the DARPA SubT finals environment. The true positive classification in (c) denotes the ratio of correctly classified noise whereas the false negative represents the ratio of noise preserved after the filtration process (i.e., the unfiltered noise) to the size of the point cloud. The data for the classification analysis (c) were obtained by spatially comparing the sensor measurements with the map of the environment provided by the organizers.

data points. With n data points within a radius κ , the computational complexity is reduced to $\mathcal{O}(n)$ from baseline $\mathcal{O}(n \log(n))$. Although to achieve optimal performance the method requires calibration to given environmental conditions, a set of reasonable parameters ($\kappa = 5$ m and $\Upsilon = 30 \text{ W m}^{-2}$ throughout many of our real-world deployments in the harshest dust conditions) suffices in the majority of applications. The performance of the dust filtering is analyzed in Figure 5 on an example UAV flight in the mine part (the dustiest zone) of the DARPA SubT finals environment.

The above method is utilizable only for sensory data containing information about the intensity of the reflected light. The rest of the sensors (RGBD cameras) are not processed, but their fusion and utilization are controlled by the amount of filtered noise in the primary LiDAR. Having the cardinality of the point sets defined in (2), the estimated amount of noise can be represented as

$$r_d = 1 - \frac{|\mathcal{P}_I|}{|\mathcal{P} \setminus \mathcal{P}_D|}, \quad (3)$$

where $r_d \in \langle 0, 1 \rangle$ is the ratio of the filtered-out observations to all the observations within the local radius κ . The RGBD cameras are then classified as unreliable (and not used in mapping or for detecting landing feasibility, as marked in Figure 4) if

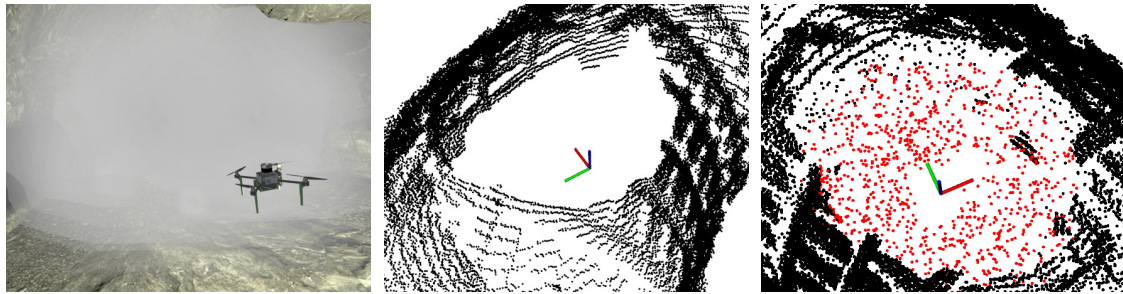
$$r_d > \lambda_d, \quad (4)$$

where $\lambda_d \in \langle 0, 1 \rangle$ is a unitless user-specified threshold. The lower value of λ_d the less amount of noise is integrated into mapping, while greater λ_d lets the connected modules handle the noise themselves. We empirically set the threshold high to $\lambda_d = 0.4$ in our final setup, since our probabilistic mapping pipeline is quite robust to the stochastic noise.

5.2.1. Detecting artificial fog in the virtual environment

The virtual competition contained a fog emitter plugin (see Figure 6) to mimic environmental perception degradation arising from observing smoke, dust, and fog. The plugin spawned a fog cloud when a robot reached the proximity of the emitter. Although our localization pipeline was able to cope with local noise, the inability to filter out the fog particles in a robust way led to a degradation of the local DenseMap, and consequently to blocking local planning which respects strict requirements on collision-free path planning. Thus, in our setup for the virtual challenge, the navigation stack did not try to enter through the fog areas but detected them, maneuvered out of them, and blocked the areas for global planning.

To detect the presence of the UAV within such a fog cloud, a discretized occupancy voxel grid is built from a set of data within a local radius (example data within a radius are shown in Figure 6c). Within this radius is compared the occupancy ratio r (number of occupied voxels to all voxels in the local grid) with maximum occupancy R given by the field of view of the sensor producing the



(a) Visualization of virtual fog in Ignition Gazebo. (b) Example 3D LiDAR data outside fog. (c) Example 3D LiDAR data inside fog (fog colored locally in red).

Figure 6. Simulated fog and its effect on sensory perception in the virtual environment. A fog cloud (a) spawns when a robot reaches its proximity. The cloud then affects the sensory inputs such that a uniform-distribution noise emerges in LiDAR data corresponding to the fog (c).

data. For each LiDAR or depth sensor, the sensor is classified as being in fog if

$$r_f > \lambda_f R, \quad (5)$$

where $\lambda_f \in \langle 0, 1 \rangle$ is a unitless multiplier converting $\lambda_f R$ to a maximal occupancy ratio threshold. The multiplier was set empirically to $\lambda_f = 0.7$ in our final setup.

For depth cameras that are not used for self-localization of the UAV, the in-fog classification solely controls whether the depth data are integrated within the mapping pipeline. However, if a localization-crucial 3D LiDAR is classified to be in fog, a backtracking behavior is triggered within the mission supervisor (see Section 11). The primary purpose of the backtracking is to prevent being stuck in fog and thus the UAV is blindly navigated out of the fog through the recent history of collision-free poses, ignoring occupied cells in the DenseMap (including possible noise from fog measurements). Lastly, detection of fog in a 3D LiDAR blocks the area in global planning.

5.3. Detecting spots safe for landing

For purposes of artifact detection and spatial mapping, the UAV carries a downward-facing RGBD camera, shown in Figure 7a. Apart from mapping the space below the UAV, the depth data of this camera are used in locating areas safe for landing throughout the UAV flight. If the sensor is marked as reliable according to (4), its depth-data frames are continuously fitted with a plane model whose coefficients are used in the binary classification of safe or unsafe landability respecting the plane-fit quality and deviation of its normal vector from the gravitational vector. The process of deciding on safe landability given a single depth-data frame is visualized in Figure 7 and described in Algorithm 1. The classification assumes that the data frame can be transformed into a gravity-aligned world coordinate frame. Inputs to Algorithm 1 are the square size of safe landing spots s (m), the minimal ratio of inliers in RANSAC plane fitting $I_{\min} \in \langle 0, 1 \rangle$, and the minimal z -axis component of unit plane-normal vector $N_{\min}^z \in \langle 0, 1 \rangle$. The square size s specifies the width of an area on which a UAV can land safely. Selection of s constraints the minimal height above the ground in which a safe landing spot detection may occur. Having a sensor with the minimal field of view θ_{\min} observing an even planar surface, the condition on line 12 in Algorithm 1 will be true for distance of the camera from the surface $d > d_{\min}$, where

$$d_{\min} = \frac{s}{2 \tan\left(\frac{\theta_{\min}}{2}\right)}. \quad (6)$$

The maximal distance d_{\max} is then given by the maximal range of the sensor. In our setup, we utilized Realsense D435 camera with $\theta_{\min} = 58^\circ$ and set $s = 1.2$ m to be marginally greater than the dimensions of our UAV platform (approximately 0.8 m wide). According to (6), the given parameters

Algorithm 1. Detecting spots safe for UAV landing in downward-facing RGBD camera.

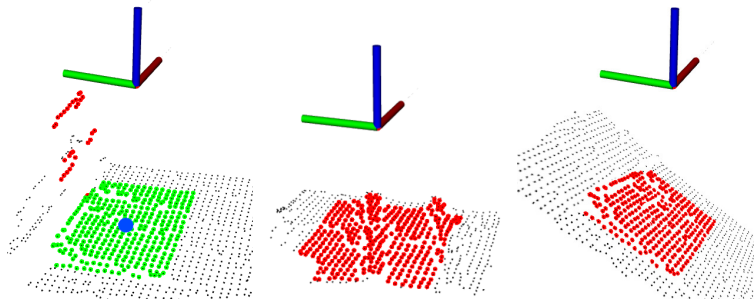
```

1: Input:
2:    $\mathcal{D}$                                      ▷ Depth-data frame in sensor coordinate frame
3: Output:
4:    $\mathcal{L}$                                      ▷ Binary classification for landing: {SAFE, UNSAFE}
5:    $\mathbf{p}_W$                                    ▷ Position of landing area in the world coordinate frame
6: Parameters:
7:    $s$                                          ▷ Square-size of safe landing spot in meters
8:    $I_{\min}$                                     ▷ Minimal ratio of inliers in plane fitting
9:    $N_{\min}^z$                                   ▷ Minimal z-axis component of the normalized plane-normal vector
10: Begin:
11:    $\mathcal{S} := \text{cropFrameAtCenter}(\mathcal{D}, s)$      ▷ Crop frame-centered square with size  $s$ 
12:   if  $\text{height}(\mathcal{S}) < s$  or  $\text{width}(\mathcal{S}) < s$  then
13:     return:  $\{\mathcal{L} = \text{UNSAFE}, \mathbf{p}_W = \text{N/A}\}$    ▷ Not safe to land: too close to the ground to decide
14:    $\mathcal{P} := \text{fitPlaneWithRANSAC}(\mathcal{S})$          ▷ Fit data with plane using RANSAC
15:   if  $\text{inliers}(\mathcal{P}) / \text{count}(\mathcal{S}) < I_{\min}$  then
16:     return:  $\{\mathcal{L} = \text{UNSAFE}, \mathbf{p}_W = \text{N/A}\}$    ▷ Not safe to land: data are not planar
17:    $\mathcal{P}_W := \text{transformToWorldFrame}(\mathcal{P})$      ▷ Transform plane to gravity-aligned frame
18:   if  $|\text{normal}(\mathcal{P}_W).z| < N_{\min}^z$  then
19:     return:  $\{\mathcal{L} = \text{UNSAFE}, \mathbf{p}_W = \text{N/A}\}$    ▷ Not safe to land: ground is too steep for landing
20:    $\mathcal{S}_W := \text{transformToWorldFrame}(\mathcal{S})$ 
21:    $\mathbf{p}_W := \text{centroid}(\mathcal{S}_W)$                  ▷ Express landing spot as the centroid of the depth data in the world
22:   return:  $\{\mathcal{L} = \text{SAFE}, \mathbf{p}_W\}$ 

```



(a) Downward-facing RGBD camera used for landability detection mounted on our UAV platform.



(b) Even planar surface: safe for landing.

(c) Nonplanar surface (rails): unsafe for landing.

(d) Uneven surface: unsafe for landing.

Figure 7. Deciding on landability of a UAV from downward-facing depth data—binary classification to safe (b) and unsafe [(c) and (d)] landing areas. In [(b)–(d)], the UAV is represented by Cartesian axes whereas the depth data are colored in black. The blue sphere in the safe classification (b) denotes the centroid of the plane inliers (colored in green) passed as a feasible landing position to LandMap (see Section 7.5).

yield the minimal distance of the sensor from the ground in detecting the landability to $d_{\min} = 1.08$ m, with $d_{\max} = 10$ m specified by the manufacturer. If the input data frame \mathcal{D} contain noise with nonplanar distribution, the condition on line 15 will classify the data as unsafe. The plane-fit and landability classification parameters $I_{\min} = 0.9$ and $N_{\min}^z = 0.7$ were found empirically for the given sensory setup. Positions classified as safe for landing on line 21 are passed to LandMap described in Section 7.5.

6. Localization

Accurate and reliable localization is critical for most other parts of the system. The ability of the reference controller to track the desired state depends largely on the quality of the available

Table 2. Approximate distribution of the environment cross-section as announced by the organizers before the Final Event.

Cross-section (m ²)	Distribution
<5	65%
5-100	20%
>100	15%

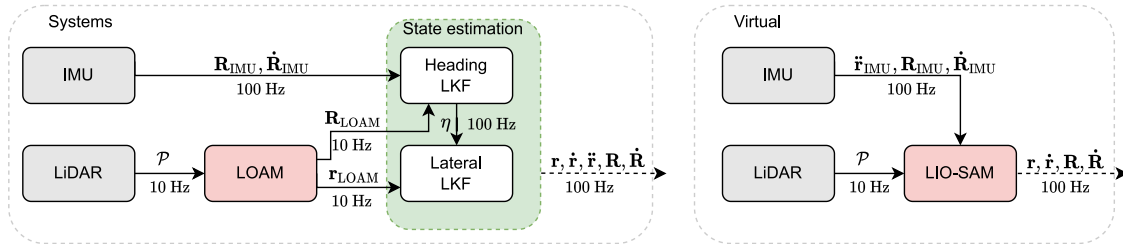


Figure 8. The diagram shows the flow of data among individual localization modules for the Systems Track (left) and Virtual Track (right). The 3D LiDAR supplies A-LOAM or LIO-SAM with the laser scans in the point cloud form \mathcal{P} . Assisted by the orientation \mathbf{R} from the IMU, A-LOAM produces a position estimate $\mathbf{r} = [x, y, z]^T$ that is fed into the *State estimation* block, which outputs the full state estimate. In the case of the virtual pipeline, the IMU data fusion is executed in LIO-SAM, and thus the state estimation module is not needed thanks to the sufficient accuracy of both lateral and heading components.

state estimate. In the narrow environments which are often present in subterranean environments (see Table 2 for cross-section distribution in the Final Event), minimizing the control error is crucial to avoid collisions. Multirobot cooperation assumes the consistency of maps created by individual robots. If the maps of two robots are not consistent due to errors in localization, the multirobot search might be suboptimal. For example, an unvisited goal can be rejected as already reached by a robot with an inconsistent map. Moreover, the localization accuracy influences the position error of a reported artifact. A UAV with localization drift over 5 m can detect and perfectly estimate the position of an artifact. Nevertheless, the report may never score a point since the position of the UAV itself is incorrect.

Our approach relies on a LiDAR sensor for localization as the laser technology proved to be more robust to the harsh conditions of the subterranean environment than the vision-based methods. We have been using LiDAR since the Tunnel circuit (Petrlík et al., 2020) where a lightweight 2D LiDAR aided by a rangefinder for measuring above ground level (AGL) height was sufficient for navigation in tunnels with a rectangular cross-section. The more vertical environment of the urban circuit required redesigning the localization system to use 3D LiDAR for navigating in 3D space (Kratky et al., 2021a).

The localization system deployed in the Final Event and presented in this manuscript builds upon the solution proposed in (Kratky et al., 2021a) and is divided into two modules: the localization algorithm and the state estimation method. Figure 8 shows the data flow in the localization pipeline. We have based the localization on the A-LOAM implementation of the LiDAR Odometry and Mapping (LOAM) algorithm (Zhang and Singh, 2014) for the Systems Track and the LiDAR Inertial Odometry via Smoothing and Mapping (LIO-SAM) (Shan et al., 2020) for the Virtual Track. Our implementation³ has been tested in a real-time UAV control pipeline throughout multiple experimental deployments as part of our preliminary works (Kratky et al., 2021a; Petracek et al., 2021) and in the DARPA SubT competition.

³github.com/ctu-mrs/aloam

6.1. A-LOAM

The A-LOAM implementation of the LOAM (Zhang and Singh, 2014) algorithm utilizes the laser scans from a multiline LiDAR to obtain its 6-DOF pose. To achieve real-time performance and accurate pose estimation at the same time, the method is divided into two parts.

The first part of the algorithm processes the incoming data at the rate of their arrival and estimates the rigid motion between the consecutive point clouds \mathcal{P}_k and \mathcal{P}_{k+1} obtained at the timestamps t_k and t_{k+1} , respectively. The process starts with finding geometric features in the input point cloud \mathcal{P}_{k+1} . The points are first sorted by the smoothness of their local neighborhood, and then those which are the least and most smooth are selected as edge and planar features, respectively. To achieve a more uniform distribution of features, the point cloud is divided into regions of the same size, and each region can contain only a limited number of edge and planar feature points. A point cannot be chosen as a feature point if there is already a feature point in its local neighborhood. A correspondence is found in \mathcal{P}_k for each edge/planar point from \mathcal{P}_{k+1} . These correspondences are then weighted by their inverse distance, and correspondences with the distance larger than a threshold are discarded as outliers. Finally, the pose transform \mathbf{T}_{k+1}^L between \mathcal{P}_{k+1} and \mathcal{P}_k is found by applying the Levenberg-Marquardt method to align the correspondences.

The second part estimates the pose of the sensor in the map \mathcal{M}_k , which is continuously built from the feature points found by the first part of the algorithm. First, \mathcal{P}_{k+1} is projected into the map coordinate system to obtain \mathcal{P}_{k+1}^W . Then, feature points are searched similarly to as is done in the first part, with the difference being that 10 times more features are found. Their correspondences are found in \mathcal{M}_k , which is divided into cubes with 10m edges. The correspondences are searched for only in the cubes intersected by the \mathcal{P}_{k+1}^W to keep the run-time bounded. The transform \mathbf{T}_{k+1}^W between \mathcal{P}_{k+1}^W and \mathcal{M}_k is obtained with the same steps as in the first part. Due to the 10-times greater amount of correspondences and search through a potentially larger map, this is a much slower process than the first part.

Thanks to the combination of both parts, the algorithm outputs the pose estimate of the rate of the LiDAR, with drift bounded by slower corrections that snap the pose to the map.

6.2. State estimation

For precise and collision-free navigation through a cluttered narrow environment, which typically appears in subterranean S&R scenarios, the control stack requires a smooth and accurate state estimate at a high rate (100 Hz). The *State estimation* module provides such an estimate through the fusion of data from Advanced implementation of LOAM (A-LOAM) and IMU. It also does this by applying filtering, rejection, and prediction techniques. We provide only a brief description of the estimation process as it is not viewed as the primary contribution and has already been presented in (Baca et al., 2021).

The state vector of the UAV is defined as $\mathbf{x} = [\mathbf{r}, \dot{\mathbf{r}}, \ddot{\mathbf{r}}, \mathbf{R}, \dot{\mathbf{R}}]^T$. The position $\mathbf{r} = [x, y, z]^T$, its first two derivatives of $\dot{\mathbf{r}}$ and $\ddot{\mathbf{r}}$, the orientation in the world frame \mathbf{R} , and the angular velocities $\dot{\mathbf{R}}$ include all the dynamics required by other onboard algorithms. Even though the position \mathbf{r} is provided by the A-LOAM algorithm, the rate of the position updates is too low for the control loop. Furthermore, the velocity and acceleration vector is not known, and must thus be estimated. A Linear Kalman Filter (LKF) of a point mass model with position, velocity, and acceleration states is employed to estimate the unknown variables at the desired rate.

While the IMU of the onboard autopilot provides the orientation \mathbf{R} , the heading⁴ η is prone to drift due to the bias of the gyroscopes in Micro-Electromechanical Systems (MEMS) IMUs. We correct this drift in a standalone heading filter, which fuses $\dot{\mathbf{R}}$ gyro measurements with A-LOAM

⁴Heading is the angle between the heading vector and the first world axis. The heading vector is the direction of the forward-facing body-fixed axis projected onto the plane formed by the horizontal axes of the world frame, as formally defined in (Baca et al., 2021).

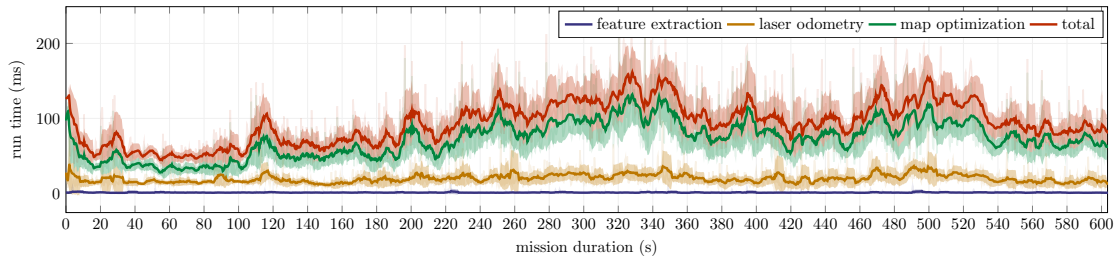


Figure 9. The computation time of the most demanding parts of the A-LOAM algorithm is plotted with respect to the time in the mission that was conducted in simulation. The total time is the sum of all three parts. The darkest colors depict moving mean, the medium dark bands represent the moving standard deviation, and raw data are shown by the lightest colors. The moving statistics are calculated over 1 s long time window. On average, the feature extraction takes 1 ms, the laser odometry 19 ms, the map optimization 91 ms, and, in total, the pose estimate is obtained in 111 ms.

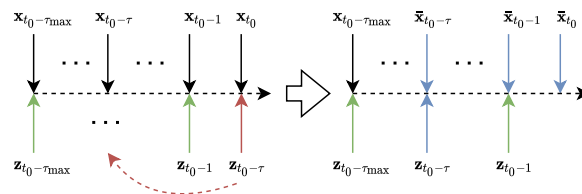


Figure 10. The left time sequence shows the situation in the filter after the arrival of delayed correction $\mathbf{z}_{t_0-\tau}$ at time t_0 . The green arrows represent corrections applied at the correct time. The delayed $\mathbf{z}_{t_0-\tau}$ would be fused at t_0 in a traditional filter, resulting in a suboptimal state estimate. However, thanks to the buffering of state and correction history, it is fused into the correct state at time $t_0 - \tau$. The states after $t_0 - \tau$ had to be recalculated to reflect the correction $\mathbf{z}_{t_0-\tau}$, which is shown by the blue color in the right time sequence.

η corrections. Corrections from the magnetometer are not considered, due to the often-occurring ferromagnetic materials and compounds in subterranean environments.

The processing of a large quantity of points from each scan and matching them into the map takes 111 ms on average (see Figure 9 for run time analysis) for the onboard Central Processing Unit (CPU). The empirical evaluation shows that the controller of the UAV becomes increasingly less stable when the state estimate is delayed for more than 300 ms. To reduce the negative effect of the delay on the control performance, we employ the time-varying delay compensation technique (Pritzl et al., 2022a). We define the delay as $\tau = t_{\mathbf{T}_{k+1}} - t_{\mathcal{P}_{k+1}}$, i.e., the time it took LOAM to compute the pose transform after receiving the point cloud from LiDAR. The core of the method is a buffer $\mathbf{Q}_{\mathbf{x}}$ containing the past states $\mathbf{x}_{(t_0-\tau_{\max}, t_0)}$, and buffer $\mathbf{Q}_{\mathbf{z}}$ having the past corrections $\mathbf{z}_{(t_0-\tau_{\max}, t_0)}$ of the filter. The length of the buffer is not fixed, but data older than the expected maximum delay τ_{\max} are discarded to keep the buffer size bounded. When a new delayed measurement $\mathbf{z}_{t_0-\tau}$ arrives at time t_0 , it is applied as a correction to the state $\mathbf{x}_{t_0-\tau}$ in $\mathbf{Q}_{\mathbf{x}}$. The corrected state $\bar{\mathbf{x}}_{t_0-\tau}$ replaces $\mathbf{x}_{t_0-\tau}$. All subsequent states $\mathbf{x}_{(t_0-\tau, t_0)}$ are discarded from $\mathbf{Q}_{\mathbf{x}}$, and replaced by the states $\bar{\mathbf{x}}_{(t_0-\tau, t_0)}$ propagated from $\bar{\mathbf{x}}_{t_0-\tau}$, using regular prediction steps of the filter with all corrections from $\mathbf{Q}_{\mathbf{z}}$. Figure 10 visualizes the sequence of performed actions. Thus we acquire a time-delay compensated state estimate which, when used in the feedback loop of the UAV controller, allows for stable flight with a delay of up to 1 s. The effect that increasing the delay has on the control error is plotted in Figure 11.

6.3. LIO-SAM

LIO-SAM (Shan et al., 2020), used in the Virtual Track approach, utilizes IMU integration on top of dual factor-graph optimization. The first factor-graph optimization is similar to the A-LOAM

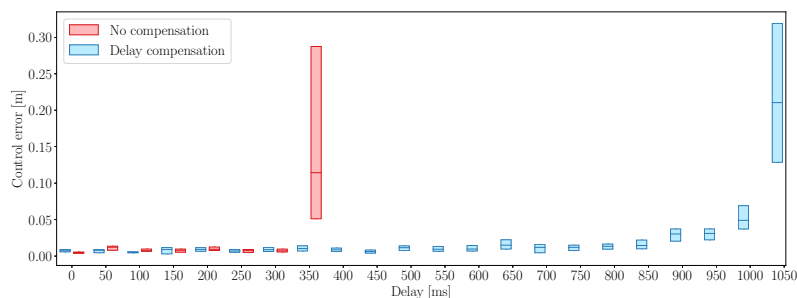


Figure 11. The box plot shows the median with lower and upper quartiles of the control error with respect to the delay of the position estimate used in the feedback loop. The data were obtained in simulation by artificially increasing the delay of ground truth position in 50 ms increments. Without compensation, the system becomes unstable after exceeding 300 ms delay, which results in oscillation-induced control error at 350 ms. The control error for the longer delay is not shown, because the high amplitude of oscillations led to a collision of the UAV. The highest delay with compensation is 1000 ms when the system has over a 5 cm control error, but is still stable. The UAV stability is lost at 1050 ms delay.

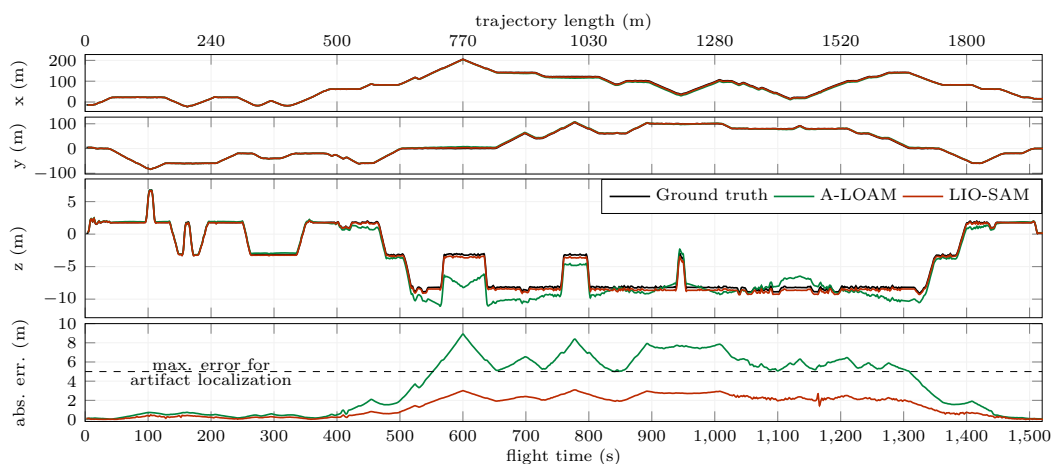


Figure 12. The performance of A-LOAM and LIO-SAM during a single flight within *Finals Prize Round World 01* (see Figure 46) of the DARPA SubT virtual environment. A-LOAM does not fuse the inertial measurements which assist LIO-SAM during LiDAR-scan matching in areas of the environment where such matching suffers from geometric degeneration, in the context of solving optimization problems. The selected environment contains a variety of narrow vertical passages where the performance of narrow-FOV LiDAR perception is limited, leading to drift in the ego-motion estimation that is clearly visible in the A-LOAM method. The LIO-SAM method was shown to achieve sufficient accuracy and low drift during long-term and arbitrary 3D navigation within a simulated environment.

mapping pipeline as it first extracts geometrical features out of raw LiDAR data and registers them to a feature map, with the motion prior given by the second optimization pipeline. The second factor-graph optimization fuses the mapping output with IMU measurements and outputs fast odometry used in the state estimation pipeline. The first graph is maintained consistently throughout the run, whereas the second graph optimization is reset periodically to maintain real-time properties.

In a simulated environment, LIO-SAM yields greater accuracy than A-LOAM for its fusion of inertial measurements with precisely modeled and known characteristics. A comparison of both the methods within the simulated environment is summarized in Figure 12. In the real world, the measurements of an IMU rigidly mounted on board a UAV contain a wide spectrum of large stochastic noise. During empirical testing, the integration method in LIO-SAM was shown to not be

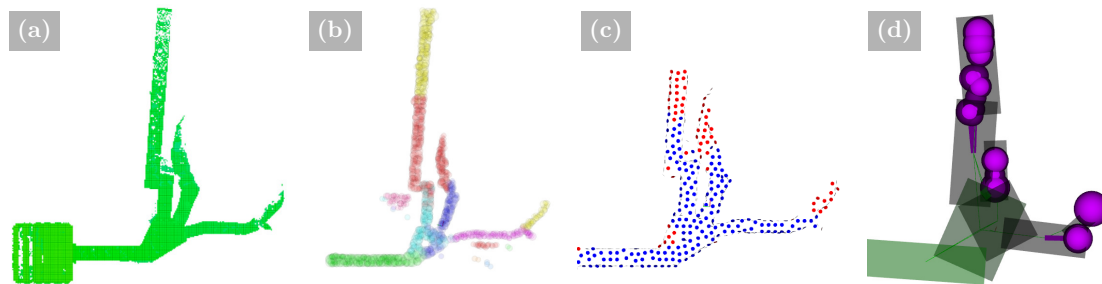


Figure 13. Top view of the used mapping structures from the intersection of the virtual Final Event map. DenseMap (a) is used for short-distance planning, SphereMap (b) for safety-aware long-distance planning, FacetMap (c) for storing surface coverage, and LTVMap (d) for compact topological information sharing among robots.

robust towards the unfiltered noise while frequency-band and pass filters induced significant time delays, destabilizing the pipeline completely. For the inability to accurately model the noise, real-world laser-inertial fusion is done manually by smoothing over a short history of past measurements (see Section 6.2).

7. Mapping

In this section, we present our approach to mapping the explored environments. As each task has specific requirements on the map properties, we designed multiple spatial representations, each of which is structured for a particular task. In particular, DenseMap (Figure 13a) is utilized for short-distance path planning; FacetMap (Figure 13b) for surface coverage tracking; SphereMap (Figure 13c) for fast and safe long-distance path planning; lightweight topological-volumetric map (LTVMap) (Figure 13d) for compressed, topological, and mission-specific information sharing between robots in low bandwidth areas; and LandMap (Figure 15) for representing feasible spots for safe UAV landing. These maps and the methods for building them are presented in this section.

7.1. DenseMap

Local information of the UAV is combined within a dense map to serve as the basis for the entire navigation stack, as described in (Kratky et al., 2021a). The map integrates information in a dense, probabilistic manner using an efficient octree structure implemented within the OctoMap (Hornung et al., 2013) library. During the map update, the data of each input modality producing spatial measurements are used to update the map with respect to the pose estimate correlating to the timestamp of the respective measurement. The data to be integrated are first cleared of any observation noise (see Section 5). The ray of each remaining spatial measurement is integrated within a discretized representation of the environment using the Bayes rule and ternary classification to the unknown, free, and occupied voxels. The output of dense mapping is convertible to other navigation representations and serves as the fundamental structure for local planning and dynamic obstacle detection.

To retain maximum information under constraints on real-time performance, the voxelization resolution is selected such that a scan insertion is processed at 5 Hz, at worst. The resolution can be locally increased if path planning demands a decrease in discretization errors. This is a useful feature for improving safety and repeatability in navigating highly narrow passages. To maintain the map structure, the local resolution is controlled by a factor n such that the local resolution equals $r/2^n$ with r being the default resolution of the dense map. In our sensory and computation setup, the default resolution is empirically set to 20 cm, reduced by a factor of $n = 2$ to 5 cm for navigating narrow passages, if required. The integrated data consist of LiDAR measurements and depth estimates of two RGBD cameras. These sensors are mounted on-board UAVs so that

the spatial observations cover roughly all directions around the robot, enabling almost arbitrary UAV-motion planning in collision-free 3D space.

7.2. SphereMap

To enable the UAV to quickly evaluate the travel time and risk caused by flying near obstacles while also pursuing any given goal, we developed a multilayer graph structure that uses volumetric segmentation and path caching, called SphereMap (Musil et al., 2022). All three layers of the SphereMap are updated near the UAV in every update iteration, which runs at approximately 2 Hz.

Path planning in the SphereMap depends on only one parameter c_R , which we call *risk avoidance*. It is used to trade path safety for path length. For long-distance planning, we disregard UAV dynamics and only take into account the path length and obstacle clearance along the path. We define the path cost between points \mathbf{p}_1 and \mathbf{p}_2 as

$$D(\mathbf{p}_1, \mathbf{p}_2) = L + c_R R, \quad (7)$$

where L is the path Euclidean length summed over all edges of the path in the sphere graph, and $R \in [0, L]$ is a risk value computed by examining the radii of the spheres along the path. For example, a path with all spheres with radii at the minimal allowed distance from obstacles would have $R = L$, and a path through open space with large sphere radii would have $R = 0$.

The lowest layer of the SphereMap is a graph of intersecting spheres, shown in Figure 13b. It is constructed by filling the free space of an obstacle k -d tree built from the DenseMap with spheres at randomly sampled points. The graph is continuously built out of intersecting spheres, and then by pruning the spheres that become unsafe or redundant. The radii of the spheres carry obstacle clearance information, which is used for path risk evaluation.

The second layer of the SphereMap is a graph of roughly convex segments of the sphere-graph. It is updated after every update of the sphere graph by creating and merging segments until every sphere in the graph belongs to a segment.

The third and last layer of the SphereMap is a navigation graph. For every two adjacent segments, we store one sphere-sphere connection, which we call a *portal* between the segments, as in (Blochliger et al., 2018). These portals form the vertices of the navigation graph. At the end of every SphereMap update iteration, we compute which paths are optimal according to the path cost from (7) between all pairs of portals of a given segment. The paths are computed only inside that given segment. If the segments are kept small (tens of meters in length), the recomputation is reasonably fast. The optimal portal-portal paths form the edges of the navigation graph. The UAV uses the navigation graph to quickly find long-distance paths between any two points in the known space by planning over the edges of the navigation graph, and then by only planning over the sphere graph in the first and last segments of the path.

7.3. FacetMap

The occupancy octree and SphereMap maps are sufficient for volumetric exploration. However, the goal of the DARPA SubT challenge was to locate artifacts, most of which could be detected only from cameras. Because the FOV of our UAVs' cameras did not cover the entire FOV of the LiDAR and depth cameras, not all occupied voxels in the occupancy map could be considered as "covered by cameras." For this reason, we developed another map, called FacetMap, illustrated in Figure 14. This map is a simple surfel map, with the facets stored in an octree structure, each having an orientation, a coverage value, and a fixed size. The FacetMap is built by computing the normals of the occupancy map at sampled occupied points, and creating facets with a set resolution if there are no existing facets with a similar normal nearby. The facets are updated (i.e., added or deleted) periodically at approximately 2 Hz in a cube of pre-defined size around the UAV.

Each facet holds a coverage value that is, for simplicity, defined as binary. A facet is marked as covered if the facet center falls into the FOV of any camera, and the ray from the camera to

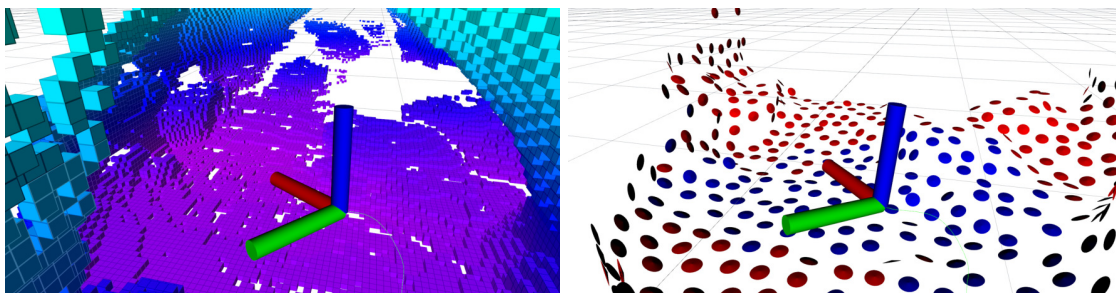


Figure 14. Illustration of the FacetMap in simulation as described in [Section 7.3](#). The map is built from the DenseMap (left) by finding normals of sampled points. The orientation of the visualization discs (right) is determined by the facet’s normal, and the color by whether the facet was covered by the UAV’s front-facing cameras or not.

the facet center is at an angle lower than a defined threshold from the facet’s normal, so as to not mark surfaces as covered if they are viewed at a very skewed angle. The angle threshold was set empirically to 78° in the competition. Angles larger than the threshold reduced the probability of successfully detecting artifacts. The covered facets stay in the map even if the underlying occupancy map shifts (e.g., when an obstacle moves). As described in [Section 8.2.3](#), one strategy used in our system uses this map to cover as much of the surface as possible while flying between volumetric exploration viewpoints. The strategy in [Section 8.2.2](#) uses this map to completely cover surfaces of a dead-end corridor before backtracking to search a different area. Coverage of entire regions of the SphereMap can also be easily computed and then stored in the LTVMap, as described in [Section 7.4](#).

7.4. LTVMap

Distributing all of the maps described in this chapter among the UAVs would be highly demanding for the communication network. As such, we have developed the lightweight topological-volumetric map (LTVMap), which combines the necessary mission-related information from the other maps and can be quickly extracted from the SphereMap and sent at any time.

This map consists of an undirected graph, where each vertex is created from a free-space segment in the original SphereMap and the edges are added for all of its adjacent segments. Each vertex holds an approximation of the segment’s shape. In our implementation, we use four DOF bounding boxes (with variable size and rotation along the vertical axis) for shape approximation, though any other shape could be used.

For cooperative exploration purposes, the frontier viewpoints (described in [Section 8.1](#)) found by a given UAV are also sent in the LTVMap, with each viewpoint being assigned an information value and segment from which the viewpoint is reachable. For surface coverage purposes, every segment in the LTVMap also holds a single numerical value representing the percentage of relevant surfaces covered in that segment. This value is computed by projecting points from the facets of the FacetMap and counting the points that fall into every segment. Further description and analysis of LTVMaps can be found in ([Musil et al., 2022](#)). These LTVMaps are shared among robots, and are used for cooperative search planning onboard UAVs, as described in [Section 8.3](#).

7.5. LandMap

As described in [Section 5.3](#), a downward-facing RGBD camera detects areas safe for landing. These areas are continuously collected within an unconnected set and stored in a sparse point-cloud manner with minimum mutual distance 5.0 m, low enough for avoiding unnecessary long paths to the nearest landing spot while keeping the LandMap memory-light even for large environment. An example of

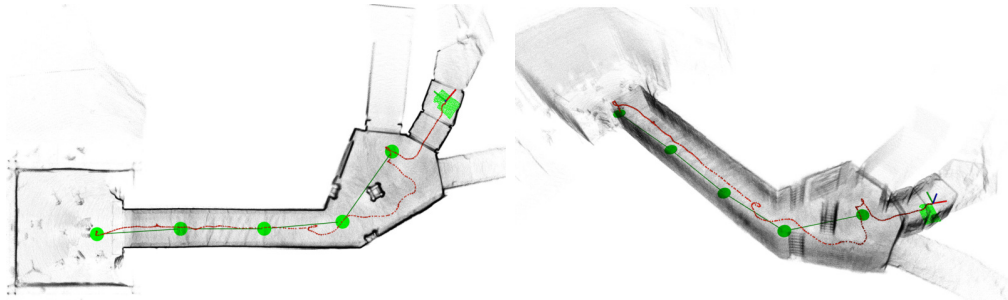


Figure 15. Example of the LandMap with resolution of 5 m built in the beginning of the DARPA SubT systems Final Event after 70s of a UAV flight. The UAV is represented by the Cartesian axes with its trajectory colored in red. The LandMap incorporates the spots classified as safe for UAV landing (green circles) which are used during the UAV homing phase of the mission to ensure safety during the landing procedure.

the LandMap is shown in Figure 15. During the homing phase of the mission, the UAV navigates to an area connected to the ground station via the communication network (see Section 11.4). After reaching this area, the UAV navigates towards a safe landing spot as indicated by the LandMap, which is closest to its current pose (see mission state machine in Figure 27). While flying towards the LandMap-selected spot, the UAV lands sooner if the ground below the UAV is classified as safe-for-landing in the current RGBD data. The landing spots previously identified as safe are, once more, verified before landing in order to ensure safety in dynamic environments. If the spot is no longer safe for landing, it is invalidated and the UAV is navigated to the next closest landing spot.

8. Autonomous search

Since communication between robots in subterranean environments can never be ensured, the UAVs in our system operate completely autonomously and only use information from other robots to update their goal decision (e.g., blocking frontiers leading to areas explored by other robots). The system can also be controlled at a very high level by the human operator, which is described in Section 11.2. This section describes the high-level search autonomy of our system.

8.1. Informative viewpoint computation and caching

For exploration purposes, the UAVs in our system do not consider the information gain along trajectories, but rather sequences of discrete viewpoints, so that we can have a unified goal representation for both local and global search planning. These viewpoints are divided into places at which a UAV could obtain some volumetric information, called *frontier viewpoints*, and the points at which a UAV could cover some not-yet-covered surfaces with its cameras, called *surface coverage viewpoints*. Each viewpoint ξ , comprising of position \mathbf{p}_ξ and heading φ_ξ , is therefore assigned some information value $I(\xi)$. In our approach, the information gain of frontier viewpoints ξ_F and surface viewpoints ξ_S is computed as

$$I(\xi_F) = c_F \frac{n_{\text{unk}}}{n_{\text{rays}}}, \quad I(\xi_S) = c_S n_{\text{unc}}, \quad (8)$$

where $n_{\text{unk}}/n_{\text{rays}}$ is the ratio of rays cast in the UAV's depth cameras' and LIDAR's FOVs that hit an unknown cell of the occupancy map before hitting an occupied one or going out of range. Similarly, n_{unc} is equal to the number of uncovered facets of the FacetMap, hit by rays that are cast in the UAV's RGB cameras' FOVs. The constants c_F and c_S are empirically tuned to alter the UAV's next viewpoint selection and hence, its behavior. Additionally, a positive or negative bias c_{FS} can be added to the information value of either function to make the UAV prefer one type of viewpoints more.

The UAV does not sample and evaluate viewpoints on-demand after reaching some viewpoint, rather it continually samples viewpoints in its vicinity at a given rate and stores them into a map of cached viewpoints. Only viewpoints that have $I(\xi)$ above some threshold, are safe, not too close to another informative viewpoint, and not blocked by mission control are stored. The viewpoints are also pruned from the map if they become uninformative or if a better viewpoint is nearby. Lastly, viewpoints that were found in a previous update and are now outside the local update box, are kept as global goals and are pruned more aggressively than the local goals. This approach continually produces a map of informative viewpoints that is denser near the UAV and sparse in the rest of the environment.

8.2. Single-UAV autonomous search planning

In our approach, the UAV can be in three states of autonomous search—*locally searching*, *traveling to goal* or *returning*, and the goal planning and evaluation is divided into local and global planning, as in (Dang et al., 2019b). In all of these states, reachability determination and path planning to any given goal is performed using the rapid long-range path finding provided by the SphereMap, described in Section 7.2. The transitions between the three states are fairly simple—if there are informative and reachable viewpoints near the UAV, the UAV is in the locally searching state and tries to always keep a sequence of two viewpoints. These are given to the trajectory planning pipeline so that the UAV doesn't stop at each viewpoint and compute the next best one. This is done by performing a local replanning of the sequence whenever the UAV is getting close to a viewpoint.

When there are no reachable viewpoints near the UAV or when new information is received from the operator or other robots, a global replanning is triggered.

The global replanning, inspired by (Dang et al., 2020b), computes paths to all stored informative viewpoints (not only in the local search box) and evaluates them. The best viewpoint is then set as a goal to the long-distance navigation pipeline described in Section 9.1. Finally, the *returning* state is triggered when the global planning does not find any reachable goals, or if the operator demands it, or if $t_{\text{home}} < c_H t_{\text{battery}}$, where t_{home} is the estimated time of flight needed to return to the base station, t_{battery} is the estimated remaining flight time, and c_H is an empirically tuned constant. The value of t_{home} is computed from the UAV's average flight speed, and a path found through the SphereMap to the base station. If there is no path to the base station, the UAV will instead try to return along a tree of visited positions, which is built specifically for this purpose, so that for example if a path is only temporarily blocked, the UAV will fly to the roadblock, and if it is removed, will continue flying to the base station. The UAV can also recover from this state, if it is returning due to having found no reachable goals, and suddenly some goals become reachable again. When the UAV gets close to the goal, it switches back to the *locally searching* state.

The reward functions used to evaluate goals govern the behavior of the UAV while searching the environment, and as such, they define the search strategy of the UAV. For simplicity, we made the local planner and global planner use the same reward function in a given strategy, with only one difference, that the local planner can add a penalty to local goals, based on the UAV's current momentum and heading, to allow for smoother local search, which is a highly simplified version of the local viewpoint tour planning done by (Zhou et al., 2021b). These strategies and their corresponding reward functions were utilized in the challenge:

8.2.1. Greedy search strategy (GS)

The chosen reward function for selecting the next best viewpoint ξ from the current UAV viewpoint ξ_{UAV} (the UAV's current position and heading) can be written as

$$R_{\text{GS}}(\xi_{\text{UAV}}, \xi) = I(\xi) - D(\xi_{\text{UAV}}, \xi), \quad (9)$$

where $I(\xi)$ is the information value of the viewpoint (described in Section 8.1) and D is the best path cost computed in the SphereMap (described in Section 7.2). This type of reward was selected for its simplicity, which allows easy debugging and tuning of UAV behavior. It is also easier to

extend this reward function to the multi-UAV cooperation reward functions in [Section 8.3](#). This reward function for controlling the next best goal selection thus depends on the constants c_F , c_S , c_{FS} described in [Section 8.1](#) and the risk-awareness constant c_R used in path planning, which can be used to tune the search based on the desired behavior. The constants c_F , c_S , c_{FS} control whether and how much the UAV prefers frontier viewpoints or surface viewpoints, while c_R is set according to the desired risk avoidance. This reward function is very simple and can take the UAV in various directions, leaving behind uncovered surfaces in faraway places. The next strategy aims to solve this.

8.2.2. Dead end inspection strategy (DEI)

A more thorough reward function can be written as

$$R_{\text{DEI}}(\xi_{\text{UAV}}, \xi) = I(\xi) - D(\xi_{\text{UAV}}, \xi) + (D(\mathbf{p}_{\text{HOME}}, \xi) - D(\mathbf{p}_{\text{HOME}}, \xi_{\text{UAV}})). \quad (10)$$

This strategy adds the difference in path costs to the base station position \mathbf{p}_{HOME} from the evaluated viewpoint and from the UAV. This greatly increases the value of viewpoints that are deeper in the environment, relative to the UAV. Using this reward function, the UAV will most likely first explore frontiers until reaching a dead-end, and then thoroughly cover surfaces from the dead end back to the base, analogous to a depth-first search.

8.2.3. Viewpoint path enhancement strategy (VPE)

The third strategy used on the UAVs is not a change of the reward function, but rather a simple way to increase surface coverage when the UAV is flying through long stretches of explored but not perfectly covered space, either in the DEI or GS strategy. If VPE is enabled and the UAV is flying to a distant goal, then we periodically take the short-distance trajectory from the local path planner (described in [Section 9](#)), sample it into multiple viewpoints, and try to perturb these viewpoints to increase surface coverage, while not increasing the flight time too much. Thus we fully utilize the agility of quadcopter UAVs, as they can easily turn from side to side while flying in a given direction.

8.2.4. Comparison of the strategies

During pre-competition testing, the three strategies mentioned above proved to be nearly identical in the total amount of covered volume and surfaces. However, there are serious differences in the overall behavior and what it means for cooperation. The GS strategy on average covers the most volume and surfaces but leaves behind many patches of surfaces or frontiers in very far-away places, due to its greedy nature. The VPE strategy just slightly alters the GS strategy to cover more surfaces in total at the cost of less explored volume but also leaves unfinished goals behind. This has generally been very useful in areas with long corridors that have a high amount of short branches leading off from the main corridor, such as in tunnels or cramped urban areas because the VPE strategy will force the UAV to peek into the corridors, but not to rigorously explore them as with the DEI strategy. The DEI strategy usually covers less space and surfaces in total, but what is most important—it does not leave uncovered and unexplored parts of the environment behind, meaning that for cooperative missions, no other UAV needs to go to that space again, as that space has been completely covered. This is essential in longer missions to ensure complete coverage of the environment.

8.3. Probabilistic cooperative search planning

Our approach to multi-UAV search planning was to make the UAVs completely autonomous and decentralized by default, while also being able to share important information and use it for their own planning. Each UAV always keeps the latest version of the LTVMap (described in [Section 7.4](#)) received from a given UAV. When a new LTVMap is received, every newest received map currently being stored onboard the UAV is updated by every other newest received map, as well as by the LTVMap constructed from the UAV's own SphereMap.

The updating is done so that the frontier viewpoints, sent along with each LTVMap, which fall into explored space in other LTVMaps, are blocked. This is difficult to do in a deterministic manner due to map drift and other inaccuracies. Therefore, we approached this problem similarly as in (Burgard et al., 2005) by gradually decreasing the reward of frontier viewpoints whenever the viewpoint falls into the explored space of any segment's bounding box in a received LTVMap. Because the LTVMap bounding boxes are a very rough approximation of the segments' true shape, this reward decreasing is weaker at the edges of the bounding boxes and strongest when the viewpoint lies deep inside the bounding box. Each frontier viewpoint in any LTVMap is assigned a likelihood $l(\xi \in V_{\text{exp}})$ to represent how likely it is that the viewpoint has already been visited by any other UAV. The $l(\xi \in V_{\text{exp}})$ of any viewpoint is the maximum of a function describing the likelihood that the point lies in a given segment's bounding box, computed over all segments of all the other received LTVMaps. This likelihood function can be selected arbitrarily; for our approach, we selected a function, which is equal to 0 outside of the segment's bounding box, and grows linearly to 1 the closer it is to the center of the bounding box. The updates of these $l(\xi \in V_{\text{exp}})$ values for a three UAV mission can be seen in Figure 17.

For a frontier viewpoint ξ_L in the UAV's local map, which has $l(\xi_L \in V_{\text{exp}}) > 0$, the reward function changes into

$$R(\xi_{\text{UAV}}, \xi_L, \mathbb{M}) = l(\xi_L \in V_{\text{exp}})R_R(\xi_{\text{UAV}}, \xi_L, \mathbb{M}) + (1 - l(\xi_L \in V_{\text{exp}}))R_L(\xi_{\text{UAV}}, \xi_L), \quad (11)$$

where R_L is the reward function defined by the employed single-UAV search strategy described in Section 8.2. This does not take into account any information from other UAVs. R_R is a reward function which takes into account other frontiers in received LTVMaps that could be reachable through ξ_L , as illustrated in Figure 16. If $l(\xi \in V_{\text{exp}}) = 0$, it means that the viewpoint does not fall into the space of any received LTVMap and the UAV only decides based on its own maps. If $l(\xi \in V_{\text{exp}}) = 1$, the viewpoint surely lies in explored space of another UAV, hence it does not

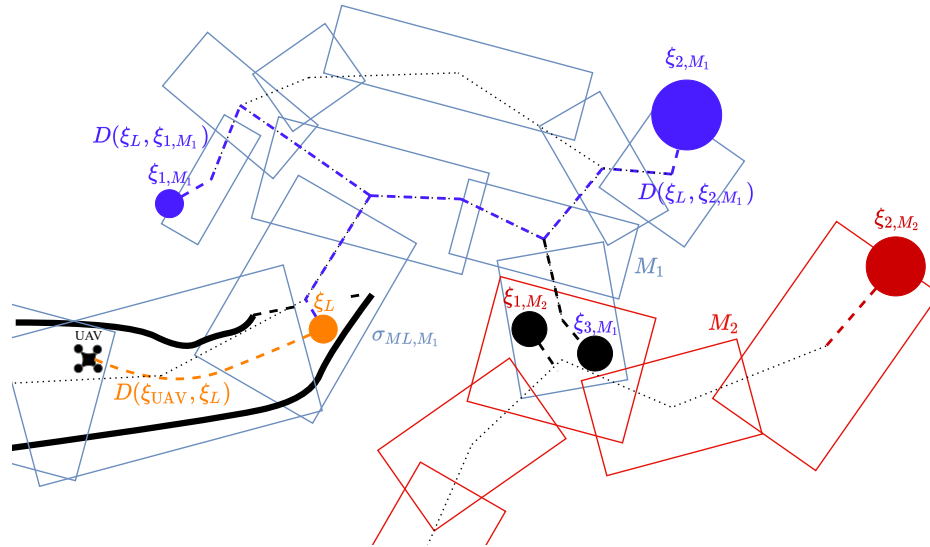


Figure 16. Diagram illustrating the computation of the cooperative exploration reward function, as described in (12). The image shows a UAV evaluating a frontier viewpoint ξ_L (orange) in its local occupancy map (black lines). The UAV has received two LTVMaps M_1, M_2 from two other UAVs. As the local map frontier ξ_L falls into one of the free space segments σ_{ML, M_1} of M_1 , it is assigned as belonging to that segment and acts as an edge in planning paths between the local map and the received map M_1 . Therefore, the frontier viewpoints $\xi_{1, M_1}, \xi_{2, M_2}$ should be reachable through ξ_L . A path to them is estimated across the centroids of the segments of M_1 . The viewpoints $\xi_{3, M_1}, \xi_{1, M_2}$ (black) are marked as having $l(\xi \in V_{\text{exp}}) = 1$, since they fall deep into the explored space of the other received map, and are therefore not considered.

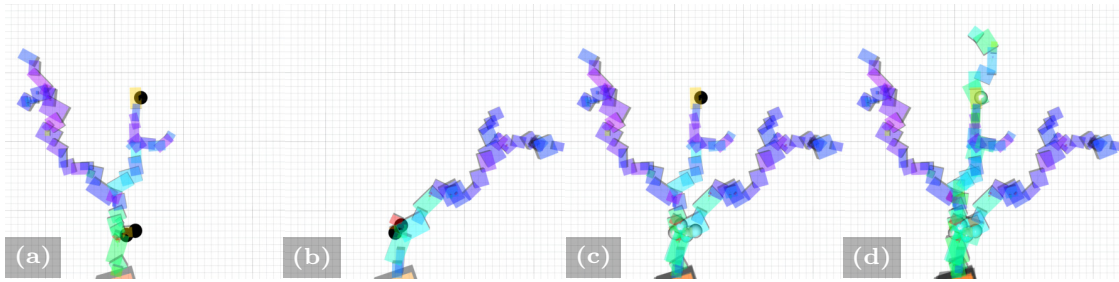


Figure 17. Illustration of LTVMap sharing and utilization during a cave exploration mission in simulation with three UAVs running the DEI strategy (described in Section 8.2.2). The heatmap color of the LTVMap segments shows surface coverage of the individual segments, with purple signifying complete coverage. The colors of the exploration viewpoints signify their $I(\xi \in V_{\text{exp}})$ value, with white having a value equal to 1 and black being 0. Image (a) shows the LTVMap sent by UAV1 after returning to communication range with the base station. This map is given to UAV2, which then launches and chooses to explore the nearest unexplored frontier in the map of UAV1. Image (b) shows the LTVMap sent by UAV2 when it is returning. Image (c) then shows how the maps are co-updated onboard UAV3, which launches after receiving the LTVMap from UAV2. The only nonexplored viewpoint remaining is in the top part of the image. Image (d) shows the maps received by the base station from all three UAVs at the end of the mission with no unexplored viewpoints remaining.

bring any volumetric information to the team, so the UAV considers whether exploring it would eventually lead it to globally unexplored viewpoints. Figure 17 illustrates how sharing the LTVMap helps UAVs to not explore already explored parts of the environment. The function R_R that achieves this behavior was designed as

$$R_R(\xi_{\text{UAV}}, \xi_L, \mathbb{M}) = \max_{M \in \mathbb{M}} \max_{\xi_R \in M} I(\xi_R) - D(\xi_{\text{UAV}}, \xi_L) - \frac{D_R(\xi_L, \xi_R, \sigma_{ML,M})}{1 - I(\xi_R \in V_{\text{exp}})}, \quad (12)$$

where \mathbb{M} is the set of all received LTVMaps, and $\sigma_{ML,M}$ is the most likely segment that ξ_L belongs to in a map M . The function D_R is a special path cost function computed as a sum of Euclidean distances of segment centers in a given map, spanning from ξ_L , through the center of $\sigma_{ML,M}$, and towards a given frontier viewpoint ξ_R . The value of D_R is also scaled by a user-defined parameter. This is done so as to increase the cost of viewpoints in received maps as there is more uncertainty about the path to these viewpoints. The division by $1 - I(\xi_R \in V_{\text{exp}})$ serves to gradually decrease the reward of exploring the viewpoint up to $-\infty$ when the viewpoint was surely explored by another UAV. Computation of this reward function is illustrated in Figure 16.

The percentage of covered surfaces inside segments received in the LTVMap is used for blocking the surface coverage viewpoints in segments, where the percentage is above a user-defined threshold. The segments with low surface coverage could be used as additional goals in a similar manner as shared frontiers in Figure 16. However, for simplicity, this was not implemented.

8.4. Autonomy robustness enhancements

One important problem is that in the case of dark and nonreflective surfaces (common in the DARPA SubT Finals course) the LiDAR beam does not return with enough energy. Such surfaces will not be marked as occupied and essentially become permanent frontiers, which means that some informative viewpoints, as defined in Section 8.2, are noninformative. To solve this, the UAV builds a map of visited positions. With time spent near a visited position, we linearly decrease the value of nearby viewpoints. After some time, the sampling is blocked near those positions completely.

Another problem arising is due to highly dynamic obstacles in the occupancy map, such as other robots, fog, or very narrow corridors where the discretization of occupancy can oscillate. As such, the reachability of a given viewpoint can oscillate. This was solved by putting a timeout on trying to reach a given viewpoint and was triggered if the UAV did not get closer to the goal within a defined

time. After this timeout, an area around the viewpoint is blocked until the end of the mission, or until a manual reset by the operator. This approach may cause the UAV to block some goals that are only temporarily blocked by another robot in narrow passages, but it was deemed preferable rather than having the UAV permanently oscillate in such passages.

The autonomy system can be easily controlled by operator commands (described in Section 11.2) which can block viewpoints in a set cylinder in space, force the UAV to explore towards some goal, or simply move to a given position and stay there. In this way, problematic situations not covered by our solution, such as organizing multiple robots in a tight corridor, can be resolved by the operator.

9. Path planning, trajectory generation and tracking

Planning collision-free paths and generating dynamically feasible trajectories is another vital component of the presented UAV system operating in a constrained environment. The sequence of waypoints (a waypoint in this context is either only a point in space, when we do not care about the heading, or a point in space and heading, for example when using the VPE strategy in Section 8.2.3, that the local planner should move the UAV through) that efficiently guides the UAV through the environment is produced by the long-distance navigation module, described in Section 9.1. Given the navigation waypoints, a computationally undemanding multistage approach is applied to obtain a trajectory lying at a safe distance from obstacles, while also respecting dynamic constraints (limits on velocity, acceleration, and jerk) and minimizing the time of trajectory following. In particular, the solution can be divided into three modules: path planning to obtain the local reference path, path processing to increase the safety margin of the path, and the trajectory generation to obtain a time-parametrized trajectory respecting the dynamic constraints of the UAV. The diagram illustrating connections and data transfer between particular modules in path planning and trajectory generation pipeline is shown in Figure 18. The long-distance path found in SphereMap, the local path found in DenseMap, the postprocessed path, and the dynamic trajectory are depicted in Figure 19.

9.1. Safety-aware long-distance navigation

When a goal, or a sequence of goals, is set to the navigation stack, the long-distance navigation module computes a path through the SphereMap, optimal according to (7). The module then keeps this path and utilizes the trajectory planning and tracking modules to follow it. This is done simply by a “carrot and stick” approach, where the trajectory planning module is given a near waypoint

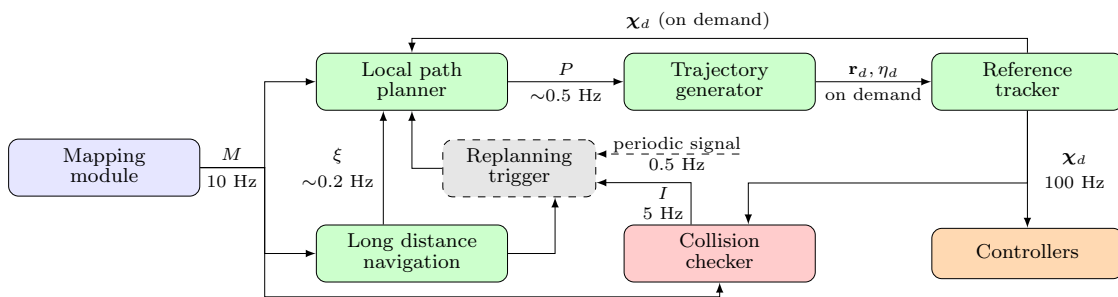


Figure 18. A diagram of the path planning and trajectory generation pipeline. Given 3D map M provided by *Mapping module*, the *Local path planner* produces path P connecting a start position for planning derived from full state reference χ_d provided by *Reference tracker*, with the viewpoints ξ supplied by *Long distance navigation* module. *Trajectory generator* produces feasible trajectory along the collision-free path P and supplies the position and heading reference (\mathbf{r}_d, η_d) to a *Reference tracker*. *Reference tracker* creates a smooth and feasible reference for the reference feedback controllers. The *Local path planner* is triggered by a new set of goals, periodic signal or by an interrupt I generated by *Collision checker* responsible for detection of collisions with respect to most recent map of the environment.

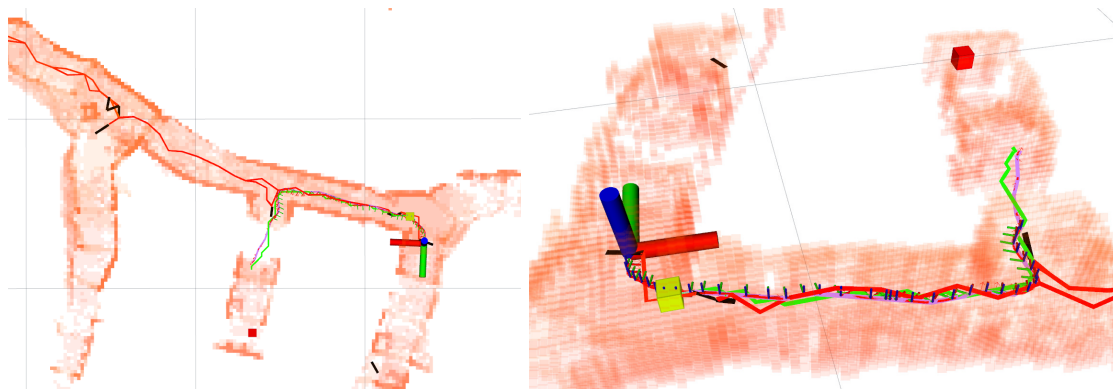


Figure 19. A two-view visualization of the path produced by all stages of the planning pipeline. The cached long-distance paths (—) between portals (—) are found in the SphereMap. A geometric path (—) is found in the DenseMap to the next waypoint given by the SphereMap. This path is then postprocessed (—) to be further away from obstacles, and a time-parametrized trajectory respecting the dynamics of the UAV is sampled (small axes). The small axes represent samples from the trajectory with constant time step, so axes further away from each other mean that the velocity of that part is higher. The current UAV pose is shown as large axes.

(approx. 20 m away from the UAV at maximum, to keep planning time short) on the path. This temporary goal waypoint is then slid across the path towards the goal.

If the trajectory planning and tracking modules cannot advance along the SphereMap path for a specified amount of time, which can be caused by a dynamic obstacle such as a rockfall, fog, or another robot, the SphereMap path following is stopped and an unreachability flag is raised. The UAV then chooses a different goal or tries to find a new path to the same goal based on the current state of mission control.

When the search planning requires the UAV to fly through multiple nearby goal viewpoints, such as when covering the surfaces in a room with cameras or when visiting multiple viewpoints while traveling and using the VPE strategy described in Section 8.2.3, the local path planning module is instead given a sequence of waypoints (containing both the goal viewpoints for surface coverage, which require heading alignment, and waypoints that do not require heading alignment and only serve to guide the local path planning). Thus the output of this module is always a sequence of one or more waypoints, which may or may not require heading alignment, and through which the local path planning module should find a path in a short time, which we can control by changing the look-ahead distance.

9.2. Local path planning

The grid-based path planner coupled with iterative path processing was adopted from (Kratky et al., 2021a) to obtain the primary reference path. The proposed approach presents a path planning and processing algorithm, which is based on the traditional A* algorithm applied on a voxel grid with several modifications to decrease the computational demands. The first modification lies in avoiding the computationally demanding preprocessing of the map representation (e.g., obstacle dilation by Euclidean distance field), which often requires more time than the actual planning on the grid. This holds true especially for shorter direct paths that leave a significant portion of the previously processed environment unexploited. For this reason, the presented approach builds a k -d tree representation of the environment which is then used to conclude the feasibility of particular cells, based on their distance to the nearest obstacle. As a result, the computational demands are partially moved from the preprocessing phase to the actual planning phase. This approach is particularly efficient in the case of paths that do not require exploiting a significant part of the environment. The second important modification is applying node pruning, similar to the jump point

search algorithm (Harabor and Grastien, 2011). This modification helps to decrease the number of unnecessarily expanded nodes. As such, it lowers the computational time required for obtaining the solution. A detailed analysis of the influence of particular modifications on the performance of the planning algorithm is provided in (Kratky et al., 2021a).

To allow the generated paths to lead through narrow passages, the limits on safety distance are set to the dimension of the narrowest opening that is supposed to be safely traversable by the UAV. However, setting this distance to a value that ensures safety in the event of the maximum possible deviation from the path caused by any external or internal source would lead to the preclusion of entering narrow passages of the environment. On the contrary, setting this distance to a minimum value without considering safety margins would increase the probability of collision along the whole path. To balance the traversability and safety of the generated path, the minimum required UAV-obstacle distance applied in the planning process is set to the lowest traversability limit, and iterative path postprocessing is applied to increase the UAV-obstacle distance in wider parts of the environment. The employed postprocessing algorithm proposed in (Kratky et al., 2021a) iteratively shifts the path towards the free part of the environment, while continually maintaining the path's connectivity. As such, this anytime algorithm increases the average UAV-obstacle distance throughout the flight, which significantly improves the reliability of the navigation with respect to imprecisions in the reference trajectory tracking.

The generated path is periodically replanned at a rate of 0.5 Hz to exploit the newly explored areas of the environment and handle dynamic obstacles. The continuous path following is achieved by using the predicted reference generated by the MPC tracker (Baca et al., 2018) to identify the starting position for the planner at time T_s in the future. Apart from the periodic replanning, the planning is also triggered by the detection of a potential collision on the prediction horizon of the trajectory reference produced by the MPC tracker. The potential collisions are checked at a rate of 5 Hz by comparing the distance of particular transition points of the predicted trajectory to the nearest obstacle in the most recent map of the environment. Depending on the time left to the time instant of a potential collision, the UAV is either requested to perform a stopping maneuver or to trigger replanning with the most up-to-date map.

9.3. Trajectory generation

The path generated by the path planning pipeline is a series of waypoints, each consisting of a 3D position and heading. A trajectory (a series of dense time-parameterized waypoints) is generated for each new path, so that the motion of the UAV satisfies translational dynamics and dynamic constraints up to the 4th derivative of position. The dynamics of the trajectory can be changed according to the current safety distance limit. However, in the Final Event, this feature was disabled, as the UAV was already constrained to 1 m s^{-1} and further slowdown would unnecessarily prolong the time spent in a narrow passage, where the risk of collision is higher. The trajectory generation system is based on the polynomial trajectory generation approach (Richter et al., 2016; Burri et al., 2015), but it was significantly extended to perform in a constrained, real-world environment (Baca et al., 2021). This approach was modified to minimize the total flight time while still satisfying the dynamic constraints. Furthermore, an iterative sub-sectioning algorithm was added to force the resulting trajectory into a feasible corridor along the original path. Moreover, a fallback solver was added to cope with invalid QP solver results caused by numerical instabilities or in case of the solver timeout. The QP solver sometimes fails to produce a feasible trajectory, e.g., by violating the dynamic constraints, or by violating the corridor constraints. In such cases, we find an alternative solution by linearly sampling each original path segment. The time duration for each segment is estimated heuristically as an upper bound using the same method as in the initialization of the polynomial trajectory generation (Baca et al., 2021). Although the trajectory produced by this method violates the dynamics in each waypoint, the underlying MPC Trajectory tracking mechanism provides smooth control reference even at these points. Most importantly, despite the fallback solution not being optimal, it is tractable and is guaranteed to finish within a fraction of the

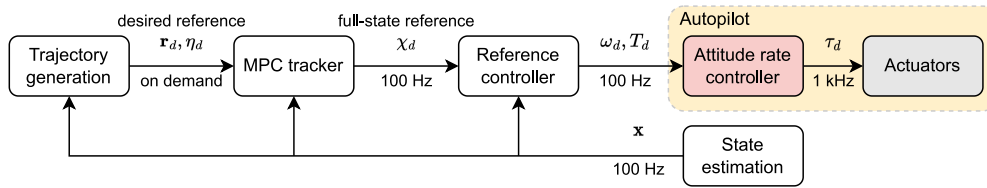


Figure 20. A diagram of the system architecture (Baca et al., 2021): *Trajectory generation* supplies the position and heading reference (r_d, η_d) to the *MPC tracker*. The *MPC tracker* creates a smooth and feasible reference for the *Reference controller*. The *Reference controller* produces the desired angular velocities and thrust (ω_d, T_d) for the embedded *Attitude rate controller*, which sets the desired speed of the motors τ_d .

time of the polynomial optimization. Finally, a dynamic initialization mechanism and a time-outing system were added to cope with the nonzero trajectory generation and path planning computation times. Even though the path planning and the trajectory generation can last for several hundreds of milliseconds, the resulting trajectory always smoothly connects to the currently tracked trajectory. Therefore, no undesired motion of the UAV is produced. The updated trajectory generation approach was released and is maintained as part of the MRS UAV System (Baca et al., 2021).

9.4. Trajectory tracking and feedback control

The low-level guidance of the UAV is provided by a universal UAV control system, as developed by the authors of (Baca et al., 2021). The onboard control system supports modular execution of UAV reference generators, feedback controllers, and state estimators. During the SubT Finals, the system exclusively utilized the geometric tracking control on $SE(3)$ (Lee et al., 2010) to follow the desired states generated by the MPC Tracker (Baca et al., 2018). First, the MPC Tracker is supplied with a time-parametrized reference trajectory, from which a smooth and feasible reference state consisting of position, velocity, acceleration, jerk, heading, and heading rate is generated by controlling a virtual model of the UAV. Second, the feedback controller minimizes the control error around the generated reference state and produces an attitude rate reference for the low-level attitude rate controller embedded in the Flight Control Unit (FCU). Figure 20 depicts the pipeline diagram of the control system with data flow among individual modules.

10. Artifact detection, localization, and reporting

Objects of interest (artifacts) in the explored area are detected visually using a CNN that processes images from several onboard RGB cameras covering the frontal, top, and bottom sectors of the UAV. The CNN detector is trained on our manually labeled dataset and outputs predicted bounding boxes and corresponding classes of the artifacts in the input images. To estimate the 3D positions of the detections, we have leveraged the onboard 3D LiDAR sensor and the mapping algorithm described in Section 7. These positions are processed by an artifact localization filter based on our previous work (Vrba et al., 2019), which fuses the information over time to filter out sporadic false positives and improve the localization precision. The artifact detection, localization, and filtering pipeline is illustrated in Figure 21.

Our approach consolidated into a similar principle of early recall and late precision proposed by (Lei et al., 2022). The CNN generates a high amount of detections to not miss any artifact at the cost of a high false positive rate. The false positives are later filtered out by the localization filter and the resulting hypotheses are further pruned by the human operator to improve the precision of the pipeline as a whole.

10.1. Artifact detection

The artifact detection is executed in parallel on image streams from all cameras at the same time, which would require a dedicated Graphical Processing Unit (GPU) onboard the UAV. Therefore,

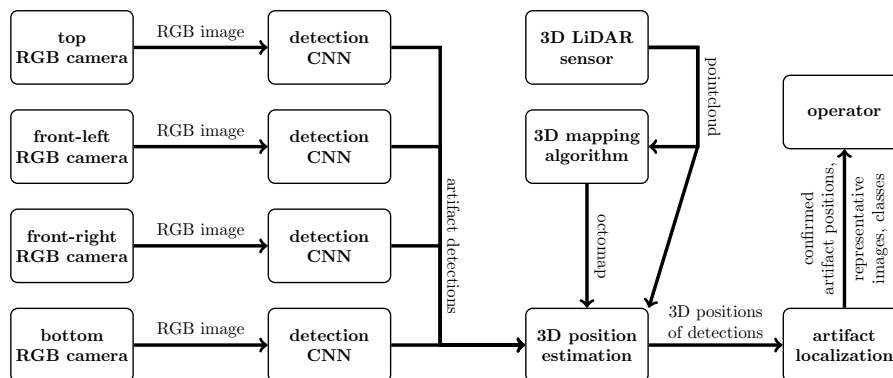


Figure 21. Schematic of the artifact detection and localization pipeline.



Figure 22. Training images containing artifacts captured by the onboard cameras in cave (a), tunnel (b), and urban (c) environments.

we have chosen the lightweight MobileNetV2 CNN (Sandler et al., 2018), in order to achieve a high detection rate and keep the load on the onboard computer as low as possible.

The CNN is running on the Intel UHD GPU that is integrated within the onboard CPU of the UAV. The integrated Intel GPU interfaces with our pipeline using the OpenVino⁵ framework. The OpenVino framework together with the Intel GPU achieves more than 5 Hz detection rate on 4 cameras in parallel but due to fixed resource allocation, we are locking the camera rates to 5 Hz. This artificial throttling of the detection rate avoids issues when the integrated GPU locks the memory resources for the CPU, which might lead to lag in the control pipeline.

The MobileNetV2 base model is modified for training using the OpenVino open-source tools. The evaluation of the model is based on the mean average precision metric (mAP) and recall. The mAP metric is a standard metric for object detection models since it provides information about how accurate the prediction is. Recall provides an understanding what is the ratio between true positive predictions and the total number of positive samples in the dataset.

The main challenge for the model is to adapt to different domains—mine, urban, and cave environments have different lighting and backgrounds (see Figure 22), which affect the detection performance. Moreover, the angle from which the images were taken is different as part of the images in the dataset were taken by ground vehicles and the rest by UAVs.

As the whole dataset was initially not available, we had to train the model incrementally whenever we gathered data from a new type of environment or camera angle to ensure we represented all cases uniformly in the training data. The incremental training was more time-efficient than retraining on the whole dataset each time new data was added. Training from scratch or checkpoints took us 2–3 days using our GPU capabilities, while incremental training produced good results in only 4–6 hours

⁵ docs.openvino.ai/latest/index.html

of training. Having the possibility to relatively quickly update the model was critical when we were doing practical experiments or hot-fixing some false-positive detections during competition days or our experimental campaign.

For training the model on a growing dataset, we used a variety of learning schedulers from the MMDetection toolbox (Chen et al., 2019). The Cosine scheduler designed by (Loshchilov and Hutter, 2016) is used for warm-restarts of the training pipeline to overcome the loss of learned features. The main challenge of transfer learning is to overcome the loss of learned distribution on the previous dataset when training the model on the new dataset (in this case the new dataset is a combination of the previous dataset and newly collected data).

In our experience, different learning rate schedulers should be used depending on the size of newly added data:

- *Cosine scheduler* (Loshchilov and Hutter, 2016) is used during clean model training on the initial dataset.
- *Cyclic scheduler* (Smith, 2015) is used when the size of new data is more than 15 % of the size of the initial dataset.
- *Step decay scheduler* is used when less than 15 % of the initial dataset size is added.

The model was trained using NVIDIA GeForce RTX 3090 video card with 24 GB of RAM, with 64 images per batch. The training size initially contained around 13 000 images and incrementally increased to 37 820 as new backgrounds and false negative samples were gradually added. Out of the 37 820 images 31 000 were labeled artifacts and 6820 were background images without any artifact to reduce the false positive rate. The train and validation split was 70 % to 30 % per training size. We open-sourced our training pipeline to facilitate replicating the achieved results by the community: github.com/ctu-mrs/darpa_subt_cnn_training. This method resulted in a score of 49.1 % mAP on the whole dataset. Such a value is acceptable on the onboard computation unit with limited resources, due to which a trade-off between accuracy and detection was necessary.

The dataset was collected using the off-the-shelf objects that were specified by the organizers, see Figure 1. The data have been recorded from the onboard cameras on the UAVs and UGVs, in particular:

- Intel RealSense D435,
- Basler Dart daA1600,
- Bluefox MLC200w.

The Basler cameras do not have an IR filter installed to maximize the amount of captured light. Altogether the dataset has 37820 images, sometimes with multiple objects in one frame. An example of images from the dataset is shown in Figure 22.

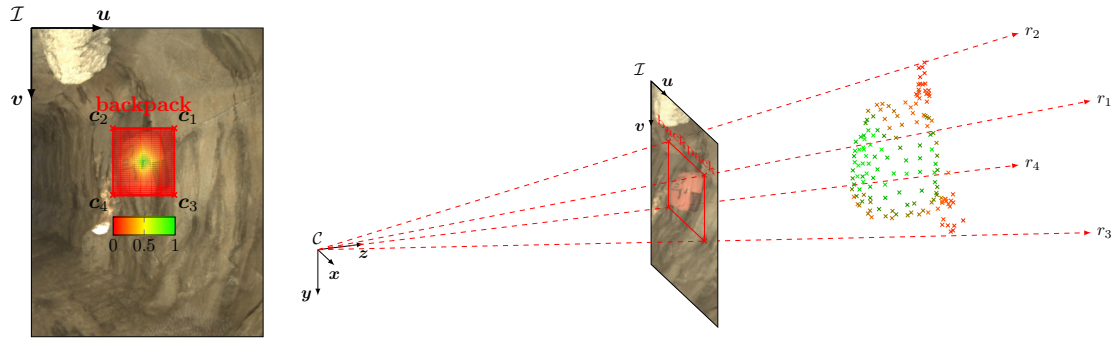
We publish the labeled detection datasets that were used for training of the neural network at github.com/ctu-mrs/vision_datasets. In addition, we also publish the tools to convert it into PASCAL VOC or COCO formats for immediate usage on most of the open-source models.

10.2. Estimation of 3D position of detections

Positions of the detected objects are estimated using data from the onboard LiDAR sensor and the mapping algorithm. Each detection is represented by four corner points $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_4$ of its bounding rectangle in the image plane of the corresponding camera, as estimated by the detector (see Figure 23a). These points are expressed as undistorted pixel coordinates in the image frame \mathcal{I} .

The mathematical projection model of the camera $f_{\text{proj}} : \mathbb{R}^3 \rightarrow \mathbb{R}^2$ is assumed to be known. In our case, we have used the standard pinhole camera model formulated as

$$k \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_u & 0 & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad (13)$$



(a) Example of an artifact detection with an overlay visualization of the sample weighting function f_w . (b) Model of the camera and the point cloud-based sampling method. Rays r_1, r_2, r_3, r_4 are projections of c_1, c_2, c_3, c_4 , respectively. Only the points within the area defined by these rays are selected. The selected points are colored based on their weight. Nonselected points are not drawn for clarity.

Figure 23. Illustration of the point sampling for 3D position estimation of detected artifacts with an example detection of a backpack.

where f_u, f_v, u_0, v_0 are parameters of the model (focal length and image center), $[x, y, z]^T$ is a 3D point in the camera coordinate frame \mathcal{C} , and u, v are distortion-free pixel coordinates in the image frame \mathcal{I} , corresponding to the 3D point (see Figure 23b for illustration). To model the distortion of the real-world camera, we have used a standard radial-tangential polynomial distortion model. It is worth noting that the output of f_{proj}^{-1} is a 3D ray and not a single point, which is represented in the model by the free scaling factor $k \in \mathbb{R}$.

The input LiDAR scan is represented as a set of 3D points $\mathcal{S} = \{\mathbf{p}_i\}$ expressed in the camera coordinate frame \mathcal{C} . The occupancy map is represented using the DenseMap data structure that is described in Section 7, and which provides a raycasting function $f_{\text{raycast}} : \mathcal{R} \rightarrow \mathbb{R}^3$ where \mathcal{R} is the set of all 3D rays. The function f_{raycast} returns the point, corresponding to the first intersection of the specified ray with an obstacle in the environment (or nothing if there is no such intersection).

The position of each detected object is estimated from a number of points that are sampled using two methods: a primary one that utilizes the latest available point cloud from the LiDAR and a secondary backup method using the latest DenseMap estimated by the mapping algorithm. The primary method is more accurate and less computationally intensive, but for artifacts lying outside of the FOV of the LiDAR scan, it may not provide enough samples for accurate 3D position estimation, which is when the secondary method is employed. For each sampled point $\mathbf{s}_i \in \mathcal{S}$, its weight w_i is calculated. The position estimate \mathbf{d} and its corresponding uncertainty covariance matrix \mathbf{Q}_d are obtained as a weighted mean of the sampled points:

$$\mathbf{d} = \sum_{i=1}^{|\mathcal{S}|} \mathbf{s}_i w_i, \quad \mathbf{Q}_d = \frac{1}{1 - \sum_{i=1}^{|\mathcal{S}|} w_i^2} \sum_{i=1}^{|\mathcal{S}|} w_i (\mathbf{s}_i - \mathbf{d}) (\mathbf{s}_i - \mathbf{d})^T, \quad (14)$$

where \mathcal{S} is the set of sampled points and the weights w_i are normalized so that $\sum_{i=1}^{|\mathcal{S}|} w_i = 1$.

The weight of a point \mathbf{s} is obtained based on the distance of its reprojection to the image coordinates $\mathbf{s}' = [s_u, s_v]^T = f_{\text{proj}}(\mathbf{s})$ from the center of the detection's bounding box $\mathbf{c}_0 = [c_u, c_v]^T$ using the function

$$f_w(\mathbf{s}', \mathbf{c}_0) = \left(1 - \frac{2|s_u - c_u|}{w_{\text{bb}}}\right)^2 \left(1 - \frac{2|s_v - c_v|}{h_{\text{bb}}}\right)^2, \quad (15)$$

where $w_{\text{bb}}, h_{\text{bb}}$ are the width and height of the bounding box, respectively. The weighting function serves to suppress points further from the center of the bounding box. This is based on our empirical observation that the center provides the most reliable estimate of the detected object's position,

Algorithm 2. Algorithm for the estimation of a detection's position and covariance.

```

1: Input:
2:    $\mathcal{D} = \{\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_4\}$ ,  $\mathbf{c}_i \in \mathbb{R}^2$                                 ▷ undistorted coordinates of the detection's bounding box
3:    $f_{\text{proj}} : \mathbb{R}^2 \rightarrow \mathcal{R}$                                           ▷ the projection model of the camera
4:    $\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{|\mathcal{P}|}\}$ ,  $\mathbf{p}_i \in \mathbb{R}^3$                         ▷ the latest point cloud from the LiDAR
5:    $f_{\text{raycast}} : \mathcal{R} \rightarrow \mathbb{R}^3$                                     ▷ the raycasting function of the occupancy map
6:    $n_{\text{desired}} \in \mathbb{N}$                                                 ▷ the desired number of sampled points
7: Output:
8:    $\mathbf{d} \in \mathbb{R}^3$                                                         ▷ estimated position of the detection
9:    $\mathbf{Q}_{\mathbf{d}} \in \mathbb{R}^{3 \times 3}$                                             ▷ covariance matrix of the position estimate
10: Begin:
11:   ▷ First, the desired number of points is sampled using the primary and secondary methods.
12:    $r_1 := f_{\text{proj}}^{-1}(\mathbf{c}_1)$ ,  $r_2 := f_{\text{proj}}^{-1}(\mathbf{c}_2)$ ,  $r_3 := f_{\text{proj}}^{-1}(\mathbf{c}_3)$ ,  $r_4 := f_{\text{proj}}^{-1}(\mathbf{c}_4)$  ▷ project the corners of the bounding box to 3D rays
13:    $\mathcal{S}_1 := \{\mathbf{p}_i \in \mathcal{P} \mid \mathbf{p}_i \text{ within the area defined by edges } r_1, r_2, r_3, r_4\}$  ▷ try to sample  $n_{\text{desired}}$  points using the primary method
14:    $n_{\text{remaining}} := \max(n_{\text{desired}} - |\mathcal{S}_1|, 0)$                             ▷ calculate the remaining number of points to be sampled
15:    $\mathcal{S}_2 := \text{sampleRectangle}(\{\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_4\}, n_{\text{remaining}}, f_{\text{proj}}, f_{\text{raycast}})$  ▷ sample any remaining points from the occupancy map
16:    $\mathcal{S} := \mathcal{S}_1 \cup \mathcal{S}_2$                                                 ▷ complement  $\mathcal{S}_1$  with the remaining points from  $\mathcal{S}_2$ 
17:   ▷ Then, the weight of each sampled point is calculated using the weighting function  $f_w$ .
18:    $\mathbf{c}_0 := \text{mean}(\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3, \mathbf{c}_4)$                                 ▷ calculate the center of the bounding box
19:   for each  $\mathbf{s}_i \in \mathcal{S}$  do
20:      $\mathbf{s}'_i := f_{\text{proj}}(\mathbf{s}_i)$                                           ▷ project the point back to the image frame  $\mathcal{I}$ 
21:      $w_i := f_w(\mathbf{s}'_i, \mathbf{c}_0)$                                         ▷ calculate its weight
22:   ▷ Finally, the position and its uncertainty are calculated as a weighted mean and covariance and returned.
23:    $\mathbf{d} := \sum_{i=1}^{|\mathcal{S}|} \mathbf{s}_i w_i$ 
24:    $\mathbf{Q}_{\mathbf{d}} = \frac{1}{1 - \sum_{i=1}^{|\mathcal{S}|} w_i^2} \sum_{i=1}^{|\mathcal{S}|} w_i (\mathbf{s}_i - \mathbf{d})(\mathbf{s}_i - \mathbf{d})^\top$ 
25:   return  $\mathbf{d}$ ,  $\mathbf{Q}_{\mathbf{d}}$ 

```

while the bounding box's corners typically correspond to the background and not the object, as illustrated in [Figure 23a](#). The whole 3D position estimation algorithm is presented in [Algorithm 2](#). The `sampleRectangle` routine used in [Algorithm 2](#) is described in [Algorithm 3](#).

The estimated positions and the corresponding covariance matrices serve as an input to the *artifact localization filter* described in the next section (refer to [Figure 21](#)). To avoid bias and numerical singularities in the filter, some special cases of the covariance calculation have to be handled. Namely, these are the following.

1. *All extracted points lie on a plane.* This happens, e.g. when all the cast rays of the secondary position estimation method intersect the same voxel of the DenseMap. The covariance matrix is then singular, which causes numerical problems with integrating the measurement.
2. *All extracted points are too close to each other.* This typically happens when the detected object is too far or too small. The covariance matrix's eigenvalues are then too small, biasing the fused position estimate of the artifact.

To avoid these problems, the estimated covariance matrix is rescaled, so that all eigenvalues conform to a specified minimal threshold before being processed by the artifact localization filter.

10.3. Artifact localization filter

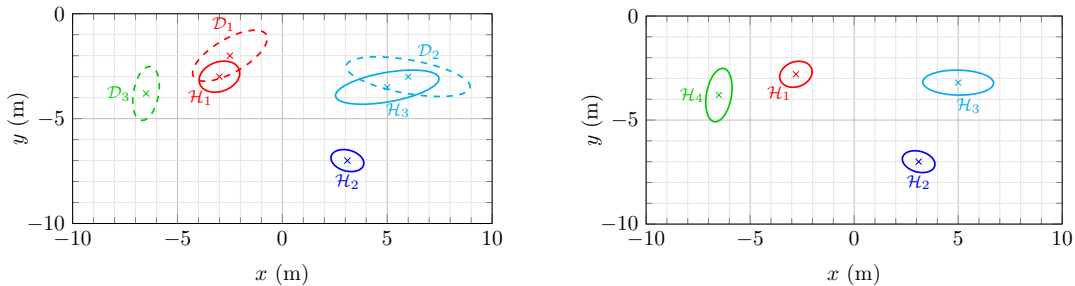
Artifact detections are filtered using an approach based on our previous work, where a multi-target tracking algorithm was employed for detection, localization, and tracking of micro aerial vehicles ([Vrba et al., 2019](#)). The filtering serves to improve the precision of the artifacts' estimated positions and to reject false positives. Only artifacts that are consistently detected multiple times with sufficient confidence are confirmed, and only the confirmed artifacts are then reported to the operator to save the limited communication bandwidth. A single step of the algorithm is illustrated in [Figure 24](#).

Algorithm 3. The `sampleRectangle` routine for sampling a number of 3D points from the occupancy map.

```

1: ▷ This routine samples points within a rectangle in the image plane  $\mathcal{I}$  by raycasting pixels on inscribed ellipses with
   an increasing radius.
2: Routine sampleRectangle:
3:   Input:
4:      $\{c_1, c_2, c_3, c_4\}$ ,  $c_i \in \mathbb{R}^2$                                 ▷ corners of the rectangle to be sampled in the image frame  $\mathcal{I}$ 
5:      $n_{\text{remaining}} \in \mathbb{N}$                                           ▷ the desired number of samples
6:      $f_{\text{proj}} : \mathbb{R}^2 \rightarrow \mathcal{R}$                             ▷ the projection model of the camera
7:      $f_{\text{raycast}} : \mathcal{R} \rightarrow \mathbb{R}^3$                        ▷ the raycasting function of the occupancy map
8:   Output:
9:      $\mathcal{S} = \{s_i\}$                                              ▷ a set of sampled points in the image frame  $\mathcal{I}$  such that  $|\mathcal{S}| \leq n_{\text{remaining}}$ 
10:  Parameters:
11:     $n_r \in \mathbb{N}$ ,  $n_\alpha \in \mathbb{N}$  ▷ number of radial sampling steps and number of circumferential steps per unit circumference
12:  Begin:
13:     $w := c_{1,u} - c_{3,u}$ ,  $h := c_{1,v} - c_{3,v}$                     ▷ calculate the width and height of the rectangle
14:     $r_{\text{step}} := 1/n_r$ 
15:    for  $r \in \{0, r_{\text{step}}, 2r_{\text{step}}, \dots, 1\}$  do
16:       $\alpha_{\text{step}} := r/n_\alpha$ 
17:       $\Delta\alpha := u$ ,  $u \sim \mathcal{U}(-\pi, \pi)$                         ▷ generate a random angular offset to avoid biasing some directions
18:      for  $\alpha \in \{0, \alpha_{\text{step}}, 2\alpha_{\text{step}}, \dots, 2\pi\}$  do
19:         $s' := [wr \cos(\alpha + \Delta\alpha)/2, hr \sin(\alpha + \Delta\alpha)/2]^T$  ▷ calculate a sample point on an ellipse
20:         $r := f_{\text{proj}}(s')$                                        ▷ project the point to a 3D ray
21:         $\mathcal{S} := \mathcal{S} \cup f_{\text{raycast}}(r)$                           ▷ find an intersection of the ray with an obstacle and add it to  $\mathcal{S}$ 
22:        if  $|\mathcal{S}| = n_{\text{remaining}}$  then
23:          return  $\mathcal{S}$ 
24:    return  $\mathcal{S}$ 

```



(a) Situation before the update step. The detections \mathcal{D}_1 and \mathcal{D}_2 are associated to the hypotheses \mathcal{H}_1 and \mathcal{H}_3 , respectively. The detection \mathcal{D}_3 is not associated to any hypothesis. The hypothesis \mathcal{H}_2 has no detection associated.

(b) Situation after the update step. The detections \mathcal{D}_1 and \mathcal{D}_2 updated the hypotheses \mathcal{H}_1 and \mathcal{H}_3 , respectively. The detection \mathcal{D}_3 initialized a new hypothesis \mathcal{H}_4 and the hypothesis \mathcal{H}_2 remained unchanged.

Figure 24. Illustration of one step of the artifact localization filter (a top-down view). Hypotheses \mathcal{H}_i are shown as covariance ellipsoids with the mean \hat{x}_i marked by an “x” symbol. Detections \mathcal{D}_i are represented in the same way using dashed lines. Associations between hypotheses and detections are highlighted using color.

The filter keeps a set of hypotheses about objects in the environment. Each hypothesis \mathcal{H} is represented by an estimate of the object’s position \hat{x} , its corresponding covariance matrix \mathbf{P} , and a probability distribution of the object’s class $p_{\mathcal{H}} : \mathcal{C} \rightarrow [0, 1]$, where \mathcal{C} is the set of considered classes. For every hypothesis \mathcal{H} , up to one detection $\mathcal{D}_{\mathcal{H}}$ is associated according to the rule

$$\mathcal{D}_{\mathcal{H}} = \begin{cases} \operatorname{argmax}_{\mathcal{D}} l(\mathcal{D} | \mathcal{H}), & \text{if } \max_{\mathcal{D}} l(\mathcal{D} | \mathcal{H}) > l_{\text{thr}}, \\ \emptyset, & \text{else,} \end{cases} \quad (16)$$

where $l(\mathcal{D} | \mathcal{H})$ is the likelihood of observing \mathcal{D} given that it corresponds to \mathcal{H} , and l_{thr} is a likelihood threshold. The associated detections are used to update the corresponding hypotheses. The detections that are not associated initialize new hypotheses.

The position estimate $\hat{\mathbf{x}}$ of a hypothesis \mathcal{H} and its covariance \mathbf{P} are updated using the Kalman filter's update equation and an associated detection $\mathcal{D}_{\mathcal{H}}$ at time step t as

$$\mathbf{K}_{[t]} = \mathbf{P}_{[t]} \mathbf{H}^{\top} (\mathbf{H} \mathbf{P}_{[t]} \mathbf{H}^{\top} + \mathbf{Q}_{\mathbf{d}[t]})^{-1}, \quad (17)$$

$$\hat{\mathbf{x}}_{[t+1]} = \hat{\mathbf{x}}_{[t]} + \mathbf{K}_{[t]} (\mathbf{d}_{[t]} - \mathbf{H} \hat{\mathbf{x}}_{[t]}), \quad (18)$$

$$\mathbf{P}_{[t+1]} = (\mathbf{I} - \mathbf{K}_{[t]} \mathbf{H}) \mathbf{P}_{[t]}, \quad (19)$$

where $\mathbf{K}_{[t]}$ is a Kalman gain, \mathbf{I} is an identity matrix, \mathbf{H} is an observation matrix (in our case, equal to \mathbf{I}), $\mathbf{d}_{[t]}$ and $\mathbf{Q}_{\mathbf{d}[t]}$ are the estimated position of $\mathcal{D}_{\mathcal{H}[t]}$ and its corresponding covariance matrix, respectively. The class probability distribution $p_{\mathcal{H}}$ is updated as

$$p_{\mathcal{H}[t+1]}(c) = \frac{n_{\text{dets}[t]} p_{\mathcal{H}[t]}(c) + p_{\mathcal{D}_{\mathcal{H}[t]}}(c)}{n_{\text{dets}[t]} + 1}, \quad (20)$$

where $c \in \mathcal{C}$ is an object's class and $n_{\text{dets}[t]}$ is the number of detections, associated to \mathcal{H} thus far.

Because the artifacts are assumed to be immobile, the Kalman filter's prediction step is not performed, which has the effect that the uncertainty of a hypothesis (represented by \mathbf{P}) can decrease without bounds. This can cause the likelihood $l(\mathcal{D} | \mathcal{H})$ of new measurements corresponding to the same object to be below the association threshold, breaking the association algorithm. To avoid this, the covariance matrix \mathbf{P} is rescaled after each update so that its eigenvalues are larger than a specified minimal value, which enforces a lower bound on the position uncertainty of the hypotheses.

10.3.1. Association likelihood

To calculate the likelihood $l(\mathcal{D}_{[t]} | \mathcal{H}_{[t]})$ of observing a detection $\mathcal{D} \equiv \{\mathbf{d}, \mathbf{Q}_{\mathbf{d}}\}$ given that it corresponds to a hypothesis $\mathcal{H} = \{\hat{\mathbf{x}}, \mathbf{P}\}$ at time step t , we use a measurement model

$$\mathbf{d}_{[t]} = \mathbf{H} \mathbf{x} + \xi_{[t]}, \quad \xi_{[t]} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{\mathbf{d}[t]}), \quad (21)$$

where \mathbf{H} is the observation matrix, \mathbf{x} is a hidden state (the real position of the artifact), $\xi_{[t]}$ is measurement noise, and $\mathcal{N}(\mathbf{0}, \mathbf{Q}_{\mathbf{d}[t]})$ denotes the Gaussian probability distribution with zero mean and covariance matrix $\mathbf{Q}_{\mathbf{d}[t]}$. Using this model, the probability density function of the expected measurement given \mathbf{x} is

$$p(\mathbf{d}_{[t]} | \mathbf{x}) = f(\mathbf{d}_{[t]} | \mathbf{H} \mathbf{x}, \mathbf{Q}_{\mathbf{d}[t]}), \quad (22)$$

where $f(\cdot | \mu, \Sigma)$ denotes the density function of the Gaussian distribution with mean μ and covariance matrix Σ .

The Kalman filter described by equations (17) to (19) can be interpreted as an estimator of the probability density of the hidden state given previous measurements. This probability density is represented as a random variable with a Gaussian distribution:

$$p(\mathbf{x} | \mathbf{d}_{[1]}, \dots, \mathbf{d}_{[t]}) = f(\mathbf{x} | \hat{\mathbf{x}}_{[t]}, \mathbf{P}_{[t]}). \quad (23)$$

The likelihood $l(\mathbf{d}_{[t]})$ of observing a new measurement $\mathbf{d}_{[t]}$ given previous measurements $\mathbf{d}_{[1]}, \dots, \mathbf{d}_{[t-1]}$ is the value of a probability density function $p(\mathbf{d} | \mathbf{d}_{[1]}, \dots, \mathbf{d}_{[t-1]})$ at $\mathbf{d}_{[t]}$. By combining equations (21) and (23), the likelihood may be expressed as

$$\begin{aligned} l(\mathbf{d}_{[t]}) &= p(\mathbf{d}_{[t]} | \mathbf{d}_{[1]}, \dots, \mathbf{d}_{[t-1]}) = \int p(\mathbf{d}_{[t]} | \mathbf{x}) p(\mathbf{x} | \mathbf{d}_{[1]}, \dots, \mathbf{d}_{[t-1]}) d\mathbf{x} \\ &= \int f(\mathbf{d}_{[t]} | \mathbf{H} \mathbf{x}, \mathbf{Q}_{\mathbf{d}[t]}) f(\mathbf{x} | \hat{\mathbf{x}}_{[t-1]}, \mathbf{P}_{[t-1]}) d\mathbf{x} \\ &= f(\mathbf{d}_{[t]} | \mathbf{H} \hat{\mathbf{x}}_{[t-1]}, \mathbf{Q}_{\mathbf{d}[t]} + \mathbf{H} \mathbf{P}_{[t-1]} \mathbf{H}^{\top}), \end{aligned} \quad (24)$$

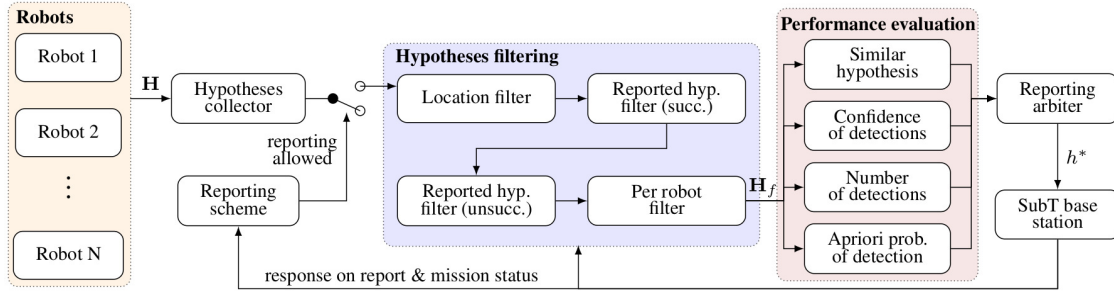


Figure 25. Illustration of the automatic reporting process from the Virtual Track.

which is the value of the probability density function of a Gaussian distribution with mean $\mathbf{H}\hat{\mathbf{x}}_{[t-1]}$ and covariance $\mathbf{Q}_{\mathbf{d}[t]} + \mathbf{H}\mathbf{P}_{[t-1]}\mathbf{H}^T$ at $\mathbf{d}_{[t]}$. This expression is used to determine the detection-to-hypothesis association at each step according to equation (16), as described in the previous section.

10.4. Arbiter for artifact reporting

In contrast to the system part of the competition, the Virtual Track requires substituting the human operator with an autonomous arbiter for artifact reporting. The main functionality of the autonomous base station resides in collecting the hypotheses from the robots and reporting the location of artifacts. The number of reports in each run is limited and usually lower than the number of hypotheses collected from all robots. Therefore, a subset of hypotheses needs to be chosen so that the expected score is maximized. The implemented reporting strategy is based on filtering the collected hypotheses by considering their location and artifact type, followed by evaluating the performance index of particular hypotheses. The entire workflow is illustrated in Figure 25.

The autonomous base station collects the hypotheses from individual robots throughout the entire run. The predefined reporting scheme specifies the maximum allowed number of reports at particular time instants of the mission. Most of the reports are saved to the last minutes of the mission when the base station holds most of the information collected from the robots. However, some reports are allowed sooner during the mission to tackle the problem of unreliable communication and prevent a failure to report all hypotheses before the time limit exceeds. When the reporting scheme allows for submitting a report, the collected hypotheses are processed to obtain the best available hypothesis h^* in a set of all collected hypotheses \mathbf{H} . First, the hypotheses are filtered using information about previous reports, their validity, location, and per robot limits on the number of reports and minimum success rate. The final set of filtered hypotheses is obtained as

$$\mathbf{H}_f = \mathbf{H} \setminus \{\mathbf{H}_{\text{area}} \cup \mathbf{H}_{\text{succ}} \cup \mathbf{H}_{\text{unsucc}} \cup \mathbf{H}_r\}, \quad (25)$$

where \mathbf{H}_{area} stands for the hypotheses located outside of the competition course, \mathbf{H}_{succ} stands for hypotheses in the vicinity of the successful reports of the same artifact class, $\mathbf{H}_{\text{unsucc}}$ contain hypotheses in the vicinity of the unsuccessful reports of the same artifact class, and \mathbf{H}_r represents the hypotheses of robots that have exceeded their own limit on reports and concurrently have a low success rate of their submitted hypotheses. The performance index for a hypothesis h_i is computed as

$$P(h_i) = \alpha p_r + \beta p_c + \gamma p_n + \delta p_a, \quad (26)$$

where the values p_r, p_c, p_n, p_a represent the percentile of particular performance indices of hypothesis h_i among all hypotheses in \mathbf{H}_f , and $\alpha, \beta, \gamma, \delta$ are the weight coefficients. The particular performance indices are related to the number of robots with a similar hypothesis (p_r), the overall confidence of the detections assigned to the hypothesis (p_c), the number of detections assigned to the hypothesis (p_n), and the apriori probability of detection of a particular object (p_a). The next hypothesis to be

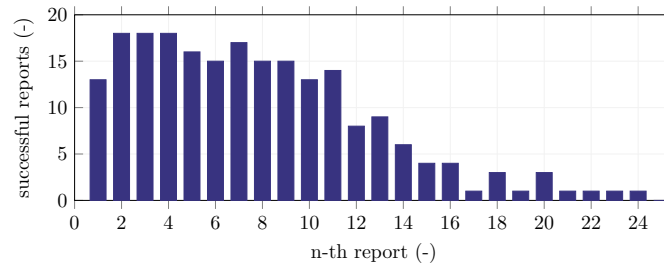


Figure 26. The distribution of successful reports over particular reporting attempts during all runs of the SubT Virtual Track Prize Round. The lower success rate of the first attempt in comparison to later attempts is caused by the early time of the first report, which was allowed 100s after the start of the mission. By this time, only a single UAV had already entered the course, and thus the number of available hypotheses to choose from was low.

reported h^* is chosen based on the following equation:

$$h^* = \arg \max_{h_i \in \mathbf{H}_f} P(h_i). \quad (27)$$

The distribution of successful reports over particular reporting attempts during all runs of the SubT Virtual Track Prize Round is shown in Figure 26. In the Systems Track, the autonomous arbiter was not used as the decision-making of the human operator regarding which hypotheses to report was superior to the autonomous arbiter, which operated based on a fixed set of rules.

11. Mission control

The proposed system is designed for fully autonomous operation, so that the rescue team can benefit from the autonomous reconnaissance of the UAV without the need for any additional personnel operating the UAV. The DARPA SubT competition reflects this requirement on autonomy by allowing only robots without human operators to enter the course. In theory, the robots could be teleoperated (Moniruzzaman et al., 2022). However, this is not scalable with the number of robots. Moreover, for teleoperation, a reliable communication link between the robot and the operator is required, but is often not available, especially deeper in the subterranean environment where impenetrable walls diminish signal propagation. Thus the correct execution of an autonomous mission relies on a state machine that governs the high-level actions of the UAV.

11.1. State machine

The state machine applied in the SubT System Finals consists of 12 fundamental states. In the first state, the status of components that are vital to the mission is checked to ensure that the mission will be accomplished. Both the software components (*localization, mapping, planning, artifact detection, artifact localization, database*) and hardware components (*LiDAR, RGB cameras, depth cameras, mobilicom unit*) are checked prior to the mission. This component health check is crucial as, while still in the staging area, any potential component failures can be addressed, but it is not possible when the UAV is already flying.

When all components are running correctly, the UAV enables the output of the reference controller, transits to *WAITING FOR TAKEOFF* state, and waits for approval from the safety operator to start the mission. The approval required to guarantee the safety of the personnel moving in the vicinity of the UAV is given by arming the UAV and transferring the control of the UAV fully to the onboard computer by toggling the Radio Controller (RC) switch. After the approval to start, the UAV waits for a specified safety timeout in the *READY FOR TAKEOFF* state while signaling the imminent takeoff by flashing LEDs. In this state, the approval can be taken back by the safety operator. After the timeout elapsed, the *PERFORMING TAKEOFF* state is entered, during which the UAV ascends until reaching the desired takeoff height.

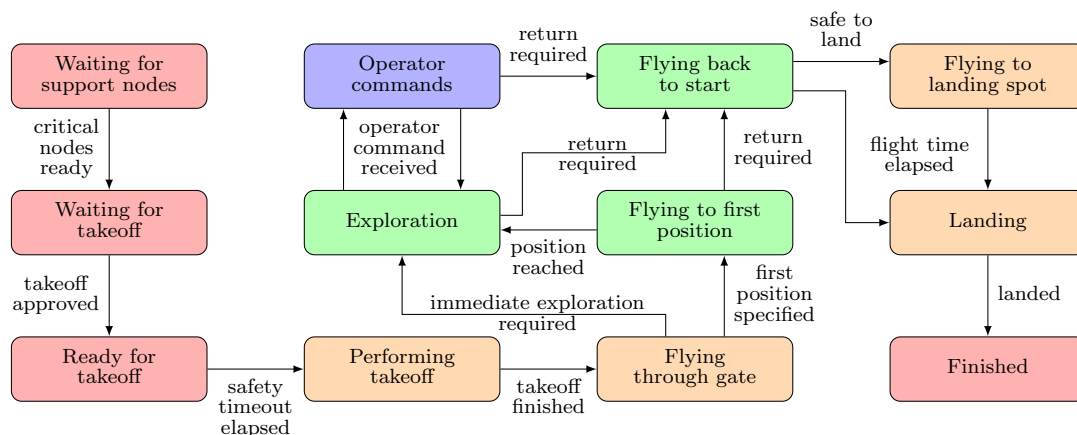


Figure 27. Simplified version of the state machine governing the autonomous mission in SubT Systems Track.

In the next state (*FLYING THROUGH GATE*), the UAV is navigated to a position inside the area to be explored. Once this position is reached, the space behind the UAV is virtually closed to prevent flight back towards the rescue personnel. If the rescuers have some prior knowledge about the environment, e.g., they see a door to which they want to send the UAV, they can optionally specify this first position to steer the UAV in that direction. After reaching this first position or if the flight to the first position is not requested, the UAV enters the *EXPLORATION* state. In this state, the UAV fulfills the primary mission goals until the upper bound of the estimated time to return is equal to the remaining flight time. Then the UAV initiates returning to the takeoff position in the state *FLYING BACK TO START*.

The return position is the takeoff position by default, but the operator can request any other position (e.g., to serve as a communication retranslation node) to which the UAV tries to return. After the position is reached, the UAV flies to the nearest safe landing spot as described in [Section 5.3](#), and the *LANDING* state is entered. The landing is also triggered when the flight time is elapsed during the *FLYING BACK TO START* or *FLYING TO LANDING SPOT* states. When the UAV lands, it enters the *FINISHED* state, in which it turns off the motors, Light-Emitting Diodes (LEDs), LiDAR, and other components except the communication modules to conserve battery power for retranslating communications.

The required communication between the UAV and its operator during the start of the mission is limited to signals provided by the RC and visual signals provided by flashing LEDs. This enables very fast deployment of the UAV that automatically starts all necessary software components once the onboard computer is powered on and provides the information about being prepared to start by a single long flash of LEDs. After that, the operator can approve the mission by the remote controller without the need for any additional communication or commanding of the UAV. Following this automated procedure, the UAVs are prepared to start one minute after the battery is plugged in.

A *FAULT* state (not shown in the simplified diagram in [Figure 27](#)) can be entered from all states in which the UAV is in the air (all states except the ones colored red in [Figure 27](#)). The *FAULT* state is entered only when it is detected that the mission cannot continue safely. In such a case, a controlled emergency landing is initiated if a position estimate is available. When a position estimate cannot be provided the emergency landing escalates into the failsafe landing, during which the UAV gradually lowers its thrust, while maintaining zero tilt. After contact with the ground is detected, the motors are turned off and the UAV is disarmed. The *FAULT* state is final, i.e., the mission cannot continue due to the failures, which triggered the transition into this state. The conditions for entering the *FAULT* state are the following.

- Data from a sensor critical for localization are not available for 1 s. This situation can happen in case of a hardware failure, detached cable, power supply failure, sensor driver bug, etc.

- The control error exceeds 2 m, which can occur with a diverging state estimate, overloaded CPU, or insufficient thrust.
- The state estimate is not available for 0.1 s, which can be caused by a bug in the state estimation module or an overloaded CPU.
- The innovation of the state estimation exceeds 2 m. Innovation is the difference between current state and a correction coming from a localization algorithm. Large innovation indicates a discrete step in the localization algorithm.
- A maximum thrust threshold of 80 % is exceeded for 1 s. This condition is triggered when a discharged battery cannot provide enough current to perform the desired motion. A faulty or older battery with many discharge cycles might struggle to provide sufficient current sooner than is the expected flight time.
- A tilt over 75° is detected, which can happen if a discrete step appears in the state estimate or when the UAV collides with an obstacle.

The state machine applied in the Virtual Track of the SubT Challenge differs only in a few states given by the specifics of the simulation environment. First, it does not contain the operator commands states that are not available in a virtual environment. Second, it contains two additional states, *BACKTRACKING* and *AVOIDING COLLISIONS*. The *BACKTRACKING* state is entered when the UAV is stuck in a fog and tries to escape from it by backtracking to the most recent collision-free poses, ignoring the occupied cells in the current map (see [Section 5.2.1](#) for details). In the *AVOIDING COLLISIONS* state, the UAV is avoiding collision with the UAVs of higher priority by stopping the lateral motion and decreasing its altitude. We have decided against using collision avoidance in the Systems Track due to the low probability of collision, and high probability of deadlocks in narrow corridors.

11.2. Operator commands

While the UAV is capable of carrying out the mission on its own in the fully autonomous mode, the operator can intervene by issuing an operator command to influence the behavior of the UAV. All operator commands can be activated only in the *EXPLORATION* state and in the operator command states, in which the UAV performs its primary goal. Allowing operator commands in other states would interfere with the takeoff, returning, and landing processes. The commands are transmitted from the operator's base station to the UAV through the available communication modalities described in [Section 11.4](#). The following commands are available for the operator:

- **Explore to position.** The operator can bias the automatic goal selection process by issuing the *Explore to position* command. After the command is received by the UAV, the currently used reward function for evaluating viewpoints is extended by a term that penalizes the Euclidean distance of the viewpoint from the desired position \mathbf{p}_D . The term added to the reward function for a viewpoint ξ is simply

$$\Delta R(\xi_{\text{UAV}}, \xi, \mathbf{p}_D) = -c_{oc} |\mathbf{p}_\xi - \mathbf{p}_D|. \quad (28)$$

Such modification of the reward function causes the viewpoints closer to the desired positions to be preferred over farther viewpoints. The assertiveness of reaching the desired position can be controlled by the coefficient c_{oc} . If this is set too high, it might force the viewpoints with a minimal distance from obstacles and low information value to be selected.

- **Plan to position.** The *Plan to position* command bypasses the viewpoint selection process and requests the planner to find a path directly to the specified position. When the requested position is not reachable, i.e., it is in an occupied or unknown space, the planner will find the path to the closest point using the Euclidean distance heuristic function. Thus this command should be used primarily for reaching an already visited position, e.g., to land there and retranslate communication from robots that are already further in the environment, or to approach a stuck robot to retrieve its data.

- **Set return position.** Under normal operation, the UAV returns to the staging area when its battery is depleted. The operator can change the return position by issuing the *Set return position* command. This can save valuable flight time of the UAV when a communication chain is already established.
- **Stop.** The operator can also halt the movement of the UAV by issuing the *Stop* command. This command is useful when the operator wants to inspect an interesting area in more detail, prevent the UAV from going into a noninformative or dangerous area, or temporarily retranslate communications. Moreover, this command is a prerequisite for calling the *Land* command.
- **Land.** It is possible to land the UAV prematurely before the end of the mission by issuing the *Land* command. The expected use case involves landing the UAV at a position advantageous for extending the communication network. Before calling the *Land* command, the *Stop* command must be called to prevent an accidental landing at an incorrect location, due to the arbitrary delay of the command sent through an unreliable network. The system does not guarantee landing at the exact specified position, as a safe landing spot is found in the vicinity of the requested position.
- **Return home.** The *Return home* command switches the UAV to the *returning* state, as defined in Section 8.2. In this state, the UAV uses the navigation module to get as close as possible to the specified return position.
- **Resume autonomy.** The last operator command cancels the behavior that was forced by previous operator commands (except *Land* and *Set return position*). This causes the UAV to resume autonomous exploration, start its return, or land (depending on the flight time left).

11.3. Operator interface

Only a single human (operator) could view the mission-specific data sent by the robots to the base station. His main task was to analyze the artifact hypotheses and report ones that seemed correct to the DARPA server to score points. He could also influence the behavior of the robots by issuing high-level operator commands (Section 11.2).

To facilitate his responsibility, each of the two tasks has a dedicated interface. Commands are issued from the RViz-based interface with each command mapped to a unique keyboard shortcut. The operator also often used live camera streams from the robots to get contextual information about the environment. This information was essential for deciding where each robot should be sent (e.g., quadrupeds to urban sections) and also for quick assessment of why a robot could be stuck.

The second interface for artifact hypotheses management is also based on RViz with a custom rqt plugin for viewing the details of each hypothesis including the image, number of detections, class probabilities, and position. These properties help the operator decide whether to send the hypothesis to the DARPA scoring server or decline it. Manual refinement of hypotheses poses is also possible by dragging them on the map.

The GUI was displayed on a semi-mobile workstation with 3 integrated displays and one external monitor standing on top of the workstation. The arrangement of the 4-displays is shown in Figure 28. From our experience, the more the operator sees without keyboard and mouse interaction, the better for his performance.

Apart from the human operator who could view all mission data, the rules also allowed the other staging area personnel to view status data. We have thus set up a diagnostics console on a computer outside the staging area. This console showed useful diagnostics information that could be relayed via voice to the human operator.

11.4. Communication

The developed system assumes an unreliable bidirectional low-bandwidth communication network with intermittent dropouts. It should be mentioned that two meshing-capable wireless technologies are used on the hardware level—2.3 GHz Mobilicom and 868 or 915 MHz motes, with details of

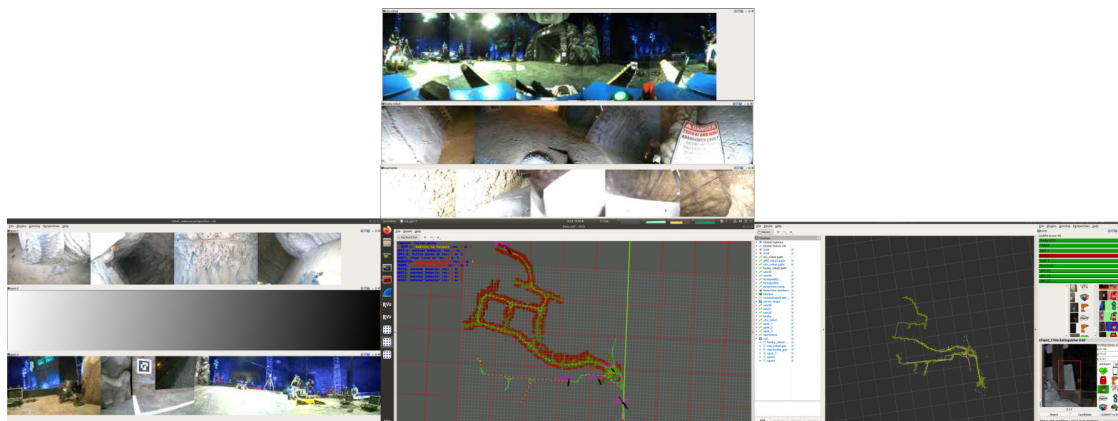


Figure 28. The operator interface display arrangement. Top screen shows live views of cameras from 3 UGVs. In bottom row, left to right, are screens with another 3 live UGV streams, the control GUI and artifact reporting GUI. The bottom left screen could also show a terminal window that was used for direct execution of scripts on the robots (as a fallback for a failure case that was not handled by the control GUI). [Figure 2](#) illustrates the physical look of the setup of the workstation with 4 displays.

both available in (Roucek et al., 2020). The motes are also dropped by UGVs as deployable range-extending battery-powered modules (Bayer and Faigl, 2020) similarly to (Ginting et al., 2021; Saboia et al., 2022) to build a communication mesh network. Our custom-made motes have lower bandwidth (100 B s^{-1}) than (Ginting et al., 2021; Saboia et al., 2022), which is compensated by sending only necessary compressed data. Moreover, bandwidth-intensive data are sent through a 1 MB s^{-1} Mobilicom network. This multimodal communication approach is robust to the failure of either Mobilicom or motes as both are able to transfer mission-critical data.

This paper focuses on high-level usage of the communication network, which is used as a black box, and as such the low-level layers of the communication protocol are not discussed.

The developed system benefits from available connections to other agents and the base station in multiple ways. First, when a robot detects an artifact, the detection with its estimated position is shared over the network instead of returning physically to the base station, thus saving time valuable for the success of the mission. Second, the individual agents can share the information about the already explored volume in the form of a topological-volumetric map (LTVMap) introduced in [Section 7.4](#). The knowledge of other agents' topological-volumetric maps penalized regions already explored by other robots, which encourages splitting of the robot team and covering a larger volume over the same time period as shown in [Figure 29](#). Third, each robot shares its position with the base station, so that the operator has an overview of where all robots are located. The operator can then influence the future behavior of any robot in the communication range by sending an operator command ([Section 11.2](#)). Last, positions of the communication nodes (breadcrumbs or landed UAVs), which form the communication network shown in [Figure 30](#), are sent to be used for returning to the communication range when the remaining flight time is low.

11.5. Calibrating global reference frame

The entire navigation system of heterogeneous robots within the CTU-CRAS-NORLAB team is decentralized under the assumption of a shared coordinate frame—the world coordinate frame O_W . To obtain the transformation of a robot's local origin within the world frame, the staging area of the competition environment provides a set of visual tags and a set of reflective markers, both with precisely known poses within the world (see the markers mounted on the entrance to the environment in [Figure 31](#)). The reflective markers are used within our 6-DOF calibration procedure in which a Leica TS16 total station is employed to measure 3D points with sub-millimeter accuracy. The origin

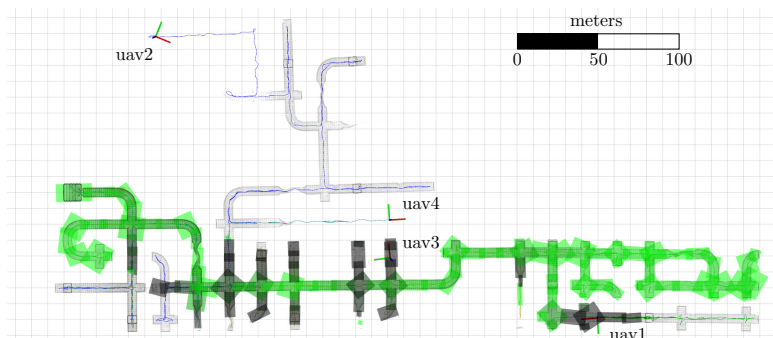


Figure 29. Example of the dispersed exploration of a tunnel system during the first run in world 1 of the virtual track. Only LTVMap from UAV1 is shown for clarity, other UAVs received this map and maps from the other UAVs. Instead of exploring again the same places as UAV1, both UAV2 and UAV4 explore previously unvisited corridors. Dark parts of LTVMap in this figure are not yet fully explored, so UAV3 flies to inspect these areas to not miss any potentially hidden artifacts.

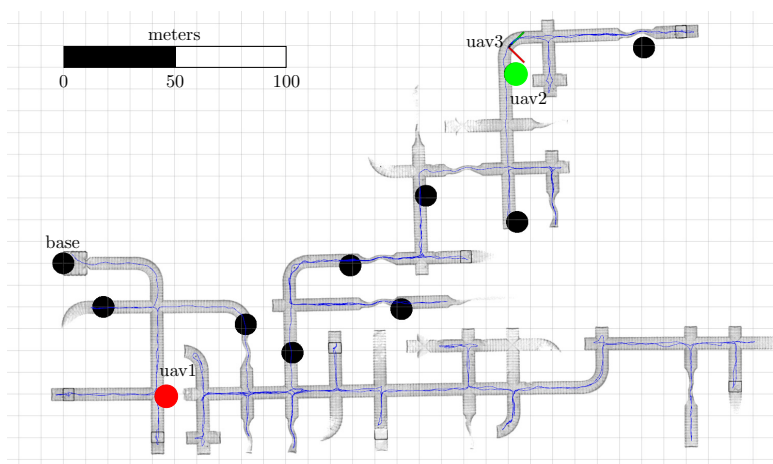


Figure 30. A communication network consisting of a base station and 8 breadcrumbs (black) deployed by the UGVs and 2 UAVs from the 3rd run in world 1 of the virtual track. UAV3 with its trajectory shown in blue could explore further thanks to the deployed communication nodes. Without the communication network, the UAV would have to return to the staging area, thus traveling additional 500 m from its final landing position.

\mathbf{T}_{TS}^W of the total station in the world is derived from measuring known in-world marker poses and used in deriving \mathbf{T}_B^W of a robot B .

To calibrate the pose of a single robot B after \mathbf{T}_{TS}^W is known, 4 known points on the robot's frame need to be measured, used in estimating \mathbf{T}_B^W , and sent to the information database (see Section 11.4) or directly to the robot. As the number of robots in the CTU-CRAS-NORLAB team deployments reached up to 9 robots per run (see Figure 31), the overhead for robots-to-world calibration decelerated the rate of robot deployments as well as limited the possibilities for quick in-situ decision-making. To speed up the calibration pipeline for UAVs with limited flight distance (and hence with greater room for calibration errors), just a single UAV A needs to be calibrated with the total station wherein the initial pose of the remaining UAVs B is estimated from on-board LiDAR data. The known transformation \mathbf{T}_A^W and pre-takeoff LiDAR data \mathbf{D}_A of a robot A are shared throughout the robots and used to estimate \mathbf{T}_B^W . The transformation \mathbf{T}_B^A is estimated by registering source LiDAR data \mathbf{D}_B onto target data \mathbf{D}_A using Iterative Closest Point (ICP) with extremely tight constraints in matching the rotation component of the transformation. The tight rotation

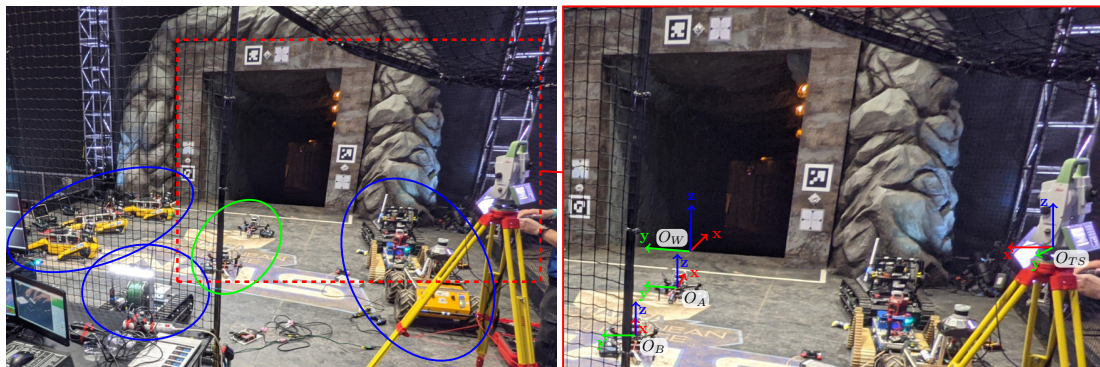


Figure 31. Example robot distribution (7 UGV robots in blue, 2 UAV robots in green) of team CTU-CRAS-NORLAB within the staging area of Systems Track environment of DARPA SubT Challenge, 2021. The Right figure highlights the reference frames of interest—the world origin O_W together with the origin of the Leica total station O_{TS} used for calibrating local robot origins O_A and O_B within the world.

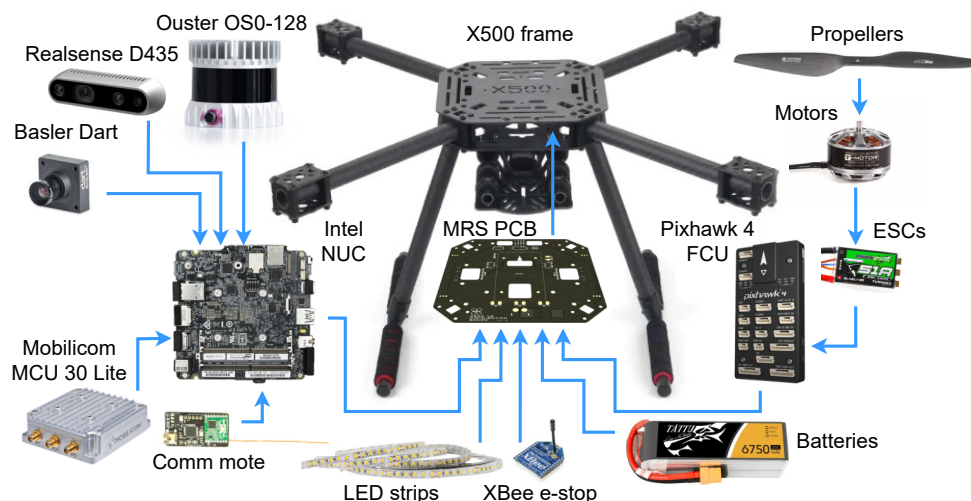


Figure 32. The interconnection of hardware components that were used is the Systems Track.

constraints are important as frame-orientation misalignments are the largest source of absolute error during deep deployments. The pose of robot B in the world is then given by $\mathbf{T}_B^W = \mathbf{T}_B^A \mathbf{T}_A^W$.

12. Hardware platform

The components of our S&R UAV were carefully selected to optimize the flight time and perception capabilities based on years of experience with building aerial robots for research (Ahmad et al., 2021), competitions (Walter et al., 2022), inspection (Silano et al., 2021), documentation (Kratky et al., 2021b) and aerial filming (Kratky et al., 2021). All platforms we have designed for diverse tasks and purposes including DARPA SubT are presented in (Hert et al., 2022).

Our platform is built upon the Holybro X500 quadrotor frame. The 500 mm frame is made entirely of carbon fiber, therefore it is stiff and light. Moreover, the arm length can be changed to accommodate different propellers. A description of all components that are mounted on the UAV frame follows. The connections of the components are depicted in Figure 32. Our team designed and manufactured a custom Printed Circuit Board (PCB) that replaced the top board of the X500 frame. This PCB (see Figure 34) supplies battery power to individual Electronic Speed Controllers (ESCs),

integrates several independent power supplies and provides a communication interface among the main computer, the Pixhawk flight controller, and MRS Modules. MRS Modules are small expansion boards that provide additional functionality and have a standardized electrical and mechanical interface. The UAV can be configured with different MRS Modules depending on the required capabilities. The PCB is connected to the main flight 4S lithium-polymer battery, which provides 14.0 V–16.8 V depending on the state of charge. The battery voltage is used to directly power the ESCs and the Intel NUC main computer. The board then integrates three independent 5 V/3 A buck converters, two to provide redundant power for the Pixhawk flight controller and one to power the MRS Modules. One 24 V/2 A boost converter is used to power the Ouster OS0-128 3D LiDAR scanner. The board has two slots for MRS Modules, one is used to control 12 V LED strips which provide illumination for the onboard RGB cameras. The second module is an interface for an XBee radio module, used as an e-stop receiver. Communication among the main computer, Pixhawk, and MRS Modules is provided by FT4232 Quad USB-UART bridge, which is integrated into the PCB. We selected MN3510 KV700 motors from T-motor and paired them with 13-inch carbon fiber propellers for large payload capacity and propulsion efficiency. The motors are driven by Turnigy Bl-Heli32 51A ESCs, as they are lightweight and easily configurable.

The 3D LiDAR was upgraded to the OS0-128 model, which features 128 scanning lines and wide 90° vertical field of view, which allows for perceiving the surroundings of the UAV in the challenging underground environments. Despite the wide coverage of the LiDAR sensor, there are still blind spots above and below the UAV when mounted horizontally. To cover these spots, we use two Intel Realsense D435 RGBD cameras, facing up and down. This enables the UAV to fly directly upwards, even in cluttered vertical shafts, without risking collision. Both of the RGBD cameras are also used for mapping and artifact detection. Additionally, the bottom facing RGBD camera is used for landing site detection. The platform is equipped with two (left and right) dedicated artifact detection cameras, the Basler Dart daA1600 with 97° horizontal FOV lens, and sufficient lighting provided by LED strips. All algorithms run on the onboard Intel NUC i7-10710U CPU with 6 physical cores and the detection CNN utilizes the integrated Intel UHD GPU.

The high-power Mobilicom MCU-30 Lite wireless communication module provides long-range connection between robots and the base station while keeping low weight of 168 g. In some topologically complex areas, even the high-power Mobilicom cannot assure reliable connection between the units, so it is supported by smaller communication motes, which are also dropped as breadcrumbs by the UGVs to improve the signal range. These motes are compact communication modules based on the RFM69HCW transceiver working at 868 MHz or 915 MHz with 100 mW transmission power and 100 B s⁻¹ data bandwidth. The performance of the motes was analyzed in the Bull Rock cave (Bayer and Faigl, 2020) and first deployed at the Urban Circuit (Roucek et al., 2020). The WiFi unit of the onboard Intel NUC computer was not used for any communication.

Finally, the large payload capacity of the UAV allowed us to extend the flight time by using a larger battery. We used two 4S 6750 mA h Li-Po batteries in parallel. Instead of a larger battery, two smaller batteries were used due to the 100 W h limit for aircraft transportation. This gave the UAV a flight time of 25 min with a total mass of 3.3 kg.

The X500 platform (Figure 33) is capable of flying in dense indoor environments, even in tight vertical shafts, while being able to localize itself with the required accuracy. It has four different cameras for artifact detection, is able to communicate and form mesh networks with other robots, and possesses a long flight time.

Furthermore, this platform was also replicated in the virtual competition with the same parameters as the physical counterpart. All of the teams except for two used the X500 platforms in the Virtual Track due to its long flight time, substantial sensor suit, and agile dynamics.

13. Technical details of hardware deployment

With a few exceptions, the components of the UAV software stack deployed in the Virtual and Systems tracks are equal, yet the available processing powers are not. The Virtual Track yields a

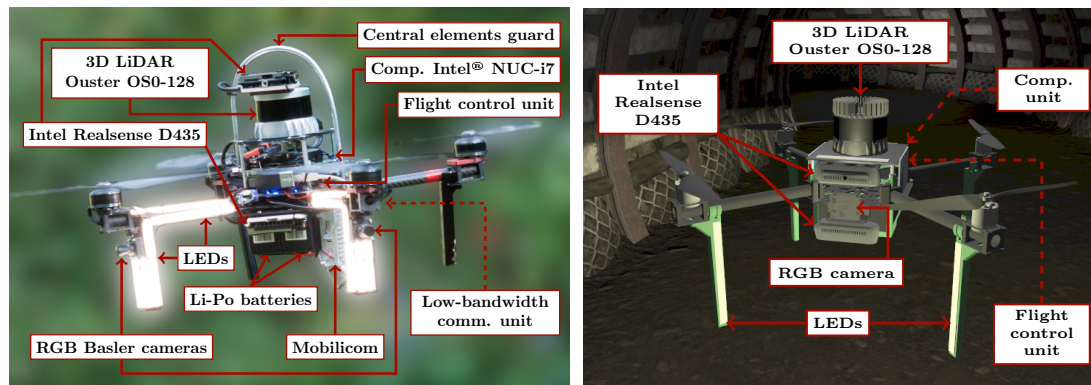


Figure 33. X500 platform used in the Systems Track (left) and Virtual Track model counterpart (right).

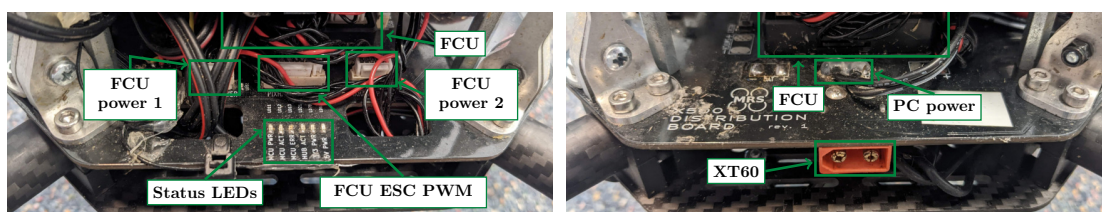


Figure 34. Custom PCB replacing the top board of the X500 frame from the front (left) and back (right). The FCU is powered by a dual redundant 5 V power supply, while the Intel NUC computer is powered directly from the 4-cell battery at 14.0V–16.8V. The battery is connected using the XT60 connector and the status of individual modules of the PCB are signaled by 6 status LEDs. The ESCs are also connected to our PCB.

low real-time simulation factor. Together with the computational capacities of each simulated robot, it provides almost unlimited computational resources for running all algorithms with any desired resolution or maximal settings. On the other hand, the simulation-to-world transition requires the algorithms to run on the onboard processing units. This imposes hard requirements on the algorithms' optimization, as well as on minimization of the amount of data transfers and their latency. These requirements force us to

- compromise between accuracy and real-time performance in the system design (i.e., cutting out global optimization in on-board running SLAM),
- ensure real-time properties for systems handling critical factors of the mission (i.e., UAV control),
- optimize the data flow and the priorities of processing order within the software stack, and
- prevent any possible deadlocks from arising from outages of both synchronous, and asynchronous data.

Ensuring real-time settings for all systems of a robotic deployment is implausible, particularly in complex robotic-research projects where the stack design must allow for the system to function as a whole under limited real-world conditions. We summarize the specific aspects of the proposed ROS-based software stack, allowing us to transfer all components to on-board processing capacities. Thus providing full decentralization within a UAV team.

Software based on ROS 1 allows for connecting components under a *nodelet manager* in order to group *nodelet* plugins. In contrast to *node* configuration, the *nodelets* under a *manager* have shared memory and do not require copying data, a tool useful particularly in the case of passing large maps within the navigation stack. Our deployment stack consists of several *managers*, each of which

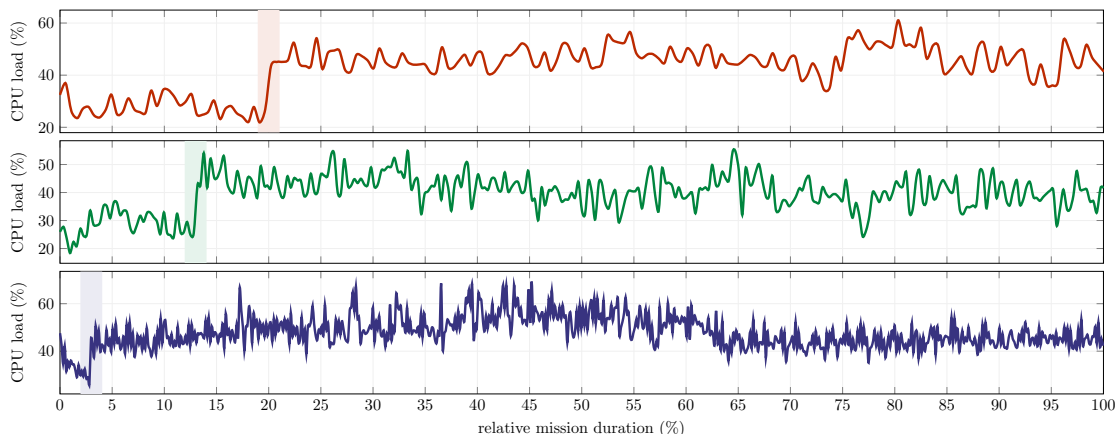


Figure 35. The CPU load of onboard computers of individual UAVs (*red, green, blue*) during the prize round of SubT Systems Track. The highlighted parts of the graph correspond to the start of processing onboard images by the object detection pipeline.

handles a distinctive part of the system. These include UAV control, preprocessing of LiDAR data and SLAM, preprocessing of RGBD data and dense mapping, navigation and path planning, and perception. The data flowing between these *managers* are copied, and thus the rate of sharing is subject to maximal reduction. To decrease the temporal and memory demands of algorithms, the resolution of input data and the output maps is decreased as much as possible within the scope and requirements of the desired application. The rate of saving data for after-mission analyses is also limited as much as possible, with no postreconstructable data being recorded at all.

In contrast to the system designs for UGV platforms, the delays in state estimation and control inputs are a critical subject for reduction. This is because excessive delays lead to destabilization of a multirotor aerial platform (see analysis on delay feasibility in [Figure 11](#)) as it is a dynamically unstable system requiring frequent feedback, even for simple hovering. The *nodelet managers* handling such critical parts of the system are prioritized at the CPU level, utilizing the negative *nice* values that prioritize the related processes during CPU scheduling. To decrease asynchronous demands on the CPU, nonprioritized components are penalized with positive *nice*. Furthermore, their scheduling is restricted on a predetermined set of threads in a multithreaded CPU. The primary subject of scheduling restriction is the perception pipeline containing a computationally heavy CNN, where static allocation reduces its asynchronous influence on the rest of the system at the cost of a limited processing rate. The effect of switching on the perception pipeline is visible in [Figure 35](#), showing the CPU load of the three deployed UAVs during the DARPA SubT Systems Track. In other validation tests, the CPU load reached up to 90% in 1500 s long missions within vast underground environments. Such an overloaded CPU results in frequent asynchronous delays, culminating to unpredictable and destructive behavior.

To limit the power consumption and hence, increase the maximum flight time, unsolicited hardware and software components can be temporarily powered off. These include switching off on-board lights in meaningless settings, disabling CNN processing when not needed, or powering off the LiDAR in the after-landing phase when the UAV is serving solely as a retranslation unit for communication.

14. System deployment

Throughout the development of the system presented in this paper, the individual components were extensively tested before integration. Deployments of the whole system were less frequent, but allowed testing the interaction of individual modules and verifying the ability to fulfill the primary objective of finding objects of interest in subterranean environments.



Figure 36. The verification of localization and perception in the following scenarios: data degraded by insufficient lighting and whirling dust (a), traversal of vertical narrow passage (b), performance in humid caves (c), multirobot exploration (d), and scalability with the environment size (e).

14.1. Continuous field verification

The S&R UAV system was continuously tested to empirically verify the correctness and reliability of the developed algorithms, strategies, and hardware. The UAVs were deployed into diverse types of environments, including historical and industrial buildings of varied levels of disintegration, in humid unstructured caves, a decommissioned underground military fortress, and vast outdoor rural areas. Some of these environments are shown in Figure 36. Such tests are critical for evaluating the performance under the stochastic influence of real-world conditions, which are typically not modeled in simulations. In particular, each perception mode is more or less degraded by ambient lighting or the lack of it, the fog with microscopic condensed droplets of water, smoke or dust particles, reflections on water or smooth surfaces, etc. The filtration of LiDAR and depth data from Section 5.2 therefore had to be tuned correctly to prevent the integration of false positives into the map, while keeping the actual obstacles. Moreover, the artifact detection system needed to work under a wide range of visibility conditions and chromatic shifts, for which it was necessary to collect artifact datasets from the mentioned environments.

14.2. DARPA SubT Final Event Systems Track

The Final Event, which was the culmination of the DARPA SubT competition, was organized in the Louisville Mega Cavern in Kentucky on September 23, 2021. The course consisted of all three environments from the previous circuits and contained all artifacts from previous events plus *the cube*, which was a new artifact for the Final Event. This section reports on the results achieved by the aerial part of the CTU-CRAS-NORLAB team. A total of 40 artifacts were distributed over 880m long course, which was divided into 28 smaller sectors to track the team's progress. Every robot starts in the staging area, from which a single corridor leads to an intersection that branches into three ways. Each of the branches leads to one of the three specific environment types (tunnel, urban, and cave).



Figure 37. All robotic platforms used in the Prize Round. (From left to right) X500, Spot, Husky, TRADR, Marmotte.

Table 3. The summary of deployed robots in the Final Event sorted by deployment times. *Operable time* means how long the robot was operable, i.e., its computers were running and it could move. *Motion time* is the time the robot was moving faster than 0.1 m s^{-1} . The row *Artifacts* shows the number of confirmed hypotheses as defined in Table 5.

Robot	Spot 1	Red	Spot 2	Marmotte	Husky	Spot 3	TRADR	Blue	Green
Locomotion	Legged	Aerial	Legged	Tracked	Wheeled	Legged	Tracked	Aerial	Aerial
Deploy time	0:20	2:00	4:00	7:20	12:40	17:32	28:20	36:00	46:30
Operable time	6:00	3:00	7:00	44:00	20:00	11:00	32:00	22:25	6:10
Motion time	3:00	2:32	2:00	6:00	5:00	9:00	4:00	15:22	4:33
Traveled	111 m	69 m	47 m	181 m	131 m	195 m	97 m	304 m	119 m
Artifacts	4	1	2	1	2	0	3	3	3
Sectors explored	4	2	2	2	2	5	4	4	4

Our team deployed a heterogeneous lineup of robots. A total of 3 legged robots (Spots), 2 tracked robots (TRADR, Marmotte), 3 aerial robots (X500), and 1 wheeled robot (Husky) robot were deployed in the Final Event Prize Round (see Figure 37). The Husky robot is a fast wheeled platform (3.6 km h^{-1} max. speed) for exploration of easy terrain. Tracked Marmotte was also fast (4 km h^{-1} max. speed) but could overcome obstacles larger than Husky could. Spots were the universal ground platform thanks to the ability to pass most terrain except slippery surfaces (max speed 5 km h^{-1}). The highest traversability among the ground platforms was offered by the tracked TRADR robot thanks to its controllable flippers (however, maximum speed is approximately 2 km h^{-1}). The primary role of the aerial robots was to explore areas unreachable by ground robots such as vertical shafts or paths blocked by obstacles. The detailed composition of the team is summarized in Table 3 together with deployment times and mission statistics. The payload of all UGVs consisted of Ouster OS0-128, 5–6× Basler Ace 2 or 1× PointGrey Ladybug 2 cameras, Xsens MTI-30 IMU, Mobilicom MCU-30 Lite, Nvidia Jetson Xavier AGX, Intel NUC 10i7FNK, LED illumination, SDC30 gas sensor, and communication motes.

14.2.1. UAV deployment summary

Three UAVs in total (*red*, *green*, and *blue*) were deployed in the 60 min long run. All UAVs used the Greedy strategy (Section 8.2.2) without VPE for the simplicity of its reward function, which made it easier to fine-tune the reward function coefficients for the competition environment and debug the UAVs' behavior. The UAV performance is summarized in Table 4 and the flight trajectories are plotted in Figure 44. The first UAV (*red*) took off just after the first UGV, arrived to the first intersection, explored 10 m of the tunnel section, returned to the intersection, flew to the cave branch where it collided with the Spot UGV (Figure 42a). The chronologically second deployed UAV was *blue*, which went into the urban branch where it traveled to a vertical alcove with a phone artifact. Then it returned to the start of the urban section, where it hovered until exhausting the battery (Figure 42c), because all viewpoints were blocked in its map corrupted by drift in the featureless urban corridor. The last deployed UAV was *green* that explored the tunnel section, where it was blocked by a dynamically added artificial wall (Figure 41). After flying through a cluttered tunnel corridor, the UAV collided with a metal rod protruding from the wall (Figure 42b).

Table 4. The mission statistics from the prize round of the Final Event. The localization accuracy was not evaluated for UAV blue. Obtaining the ground truth position using scan matching would have been extremely strenuous due to the degenerate geometry of LiDAR scans in the urban tunnel. This degeneracy also caused the onboard localization to drift several meters.

UAV	Red	Green	Blue
Localization accuracy:			
avg max error in translation (m)	0.38 0.63	0.97 2.66	-
avg max error in heading (°)	0.64 4.06	1.48 5.37	-
Safety clearance	0.4 m	0.11 m	0.21 m
Landing cause	Collision with UGV	Collision with a metal rod protruding from the wall	Depleted battery after being trapped in degraded map

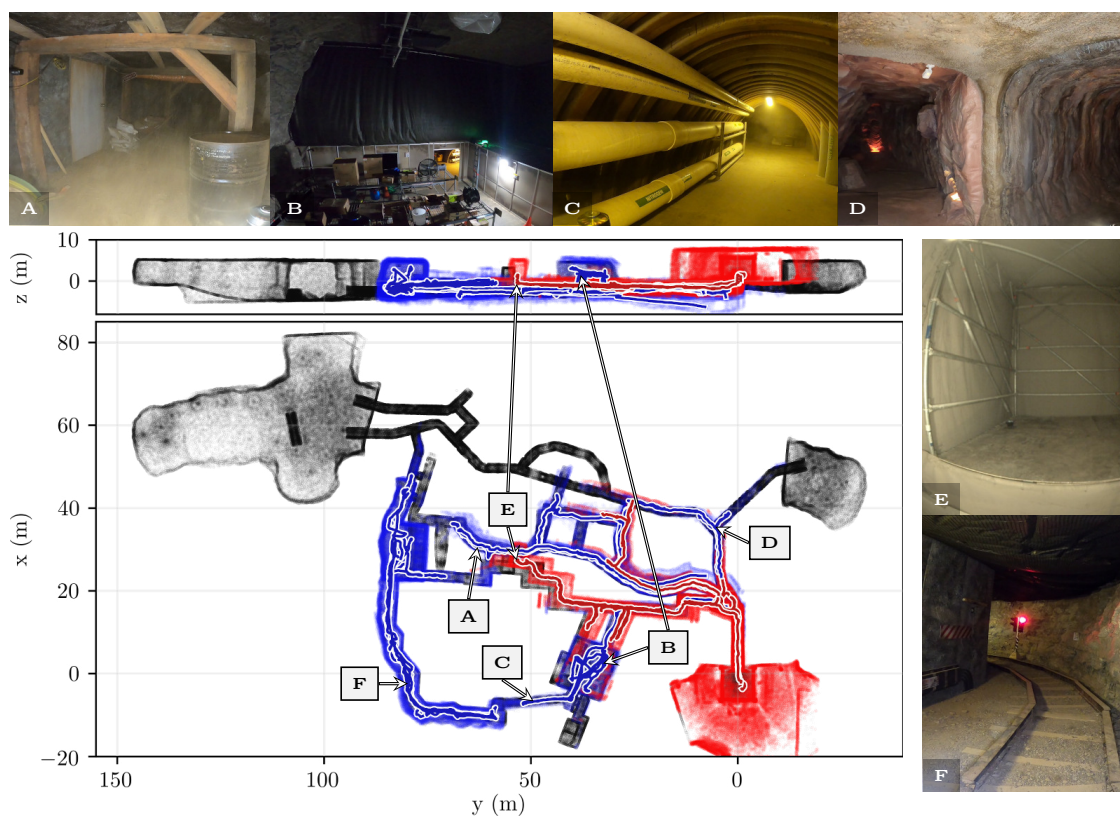


Figure 38. UAV trajectories and on-board-built maps of the environment from all flights during the prize round (colored in red) and the postevent testing (colored in blue) overlaid over the ground truth map (colored in black). The photos from on-board camera highlight the diversity and the narrow confines of the environment.

The maps and the trajectories of all our UAV flights during the prize round and the postevent testing are shown in Figure 38, together with summary of the mapping errors from these flights in Figure 39. The distance of the UAVs from the nearest obstacle during all flights in the prize round are shown in Figure 40.

14.2.2. Artifact detection discussion

The performance of the artifact detection and localization system is summarized in Table 6, and the number of artifacts detected by each UAV in Table 5. A total of seven artifacts appeared in

52 · Petrlík et al.

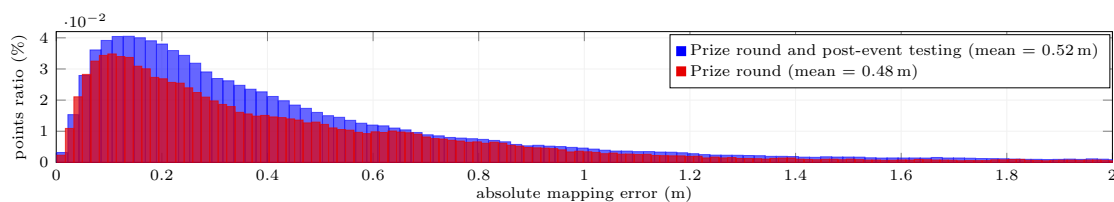


Figure 39. Distribution of mapping errors throughout the prize round and the postevent testing flights (colored in red and in blue in Figure 38) of DARPA SubT. The absolute mapping error denotes the distance between the ground truth map and concatenation of DenseMaps built with resolution of 20 cm on-board during particular UAV flights. The error metric is the Euclidean distance between a point from the on-board maps to its closest point in the ground truth map.

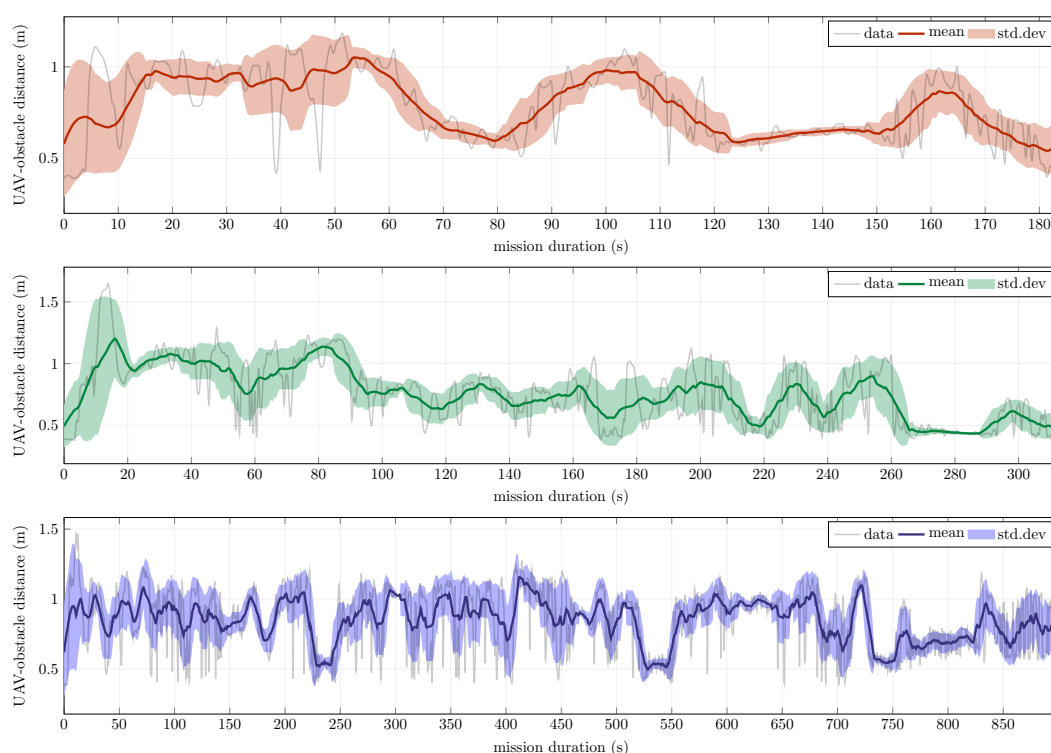


Figure 40. Distance between the center of the UAVs and the measured nearest obstacle during the prize round of the SubT Systems Track. The moving mean and standard deviation are computed over a 10 s long time window.

the camera images, and six artifacts were detected by the CNN. The detections with estimated bounding boxes from all UAVs are shown in Figure 45. The survivor *s2* was seen in three frames of the bottom camera. However, only a small part of the survivor sleeve was visible and the images were further degraded by motion blur, as can be seen in Figure 43. Thus the CNN did not manage to detect the artifact. From the six detections, the cellphone artifact *p1* was detected only on one image frame when the UAV *blue* peeked into the vertical shaft in the urban part. However, as explained in Section 10, a total of four detections are necessary to create a hypothesis and to confirm the position, and thus this single detection was discarded. Another missed point was the survivor *s1*, which was detected and localized within the 5 m limit, but the artifact was labeled as a cube instead of a survivor. The hypothesis was merged with a high number of false positives and, consequently, the correct image was not sent to the operator, who could not determine the correct class to report.

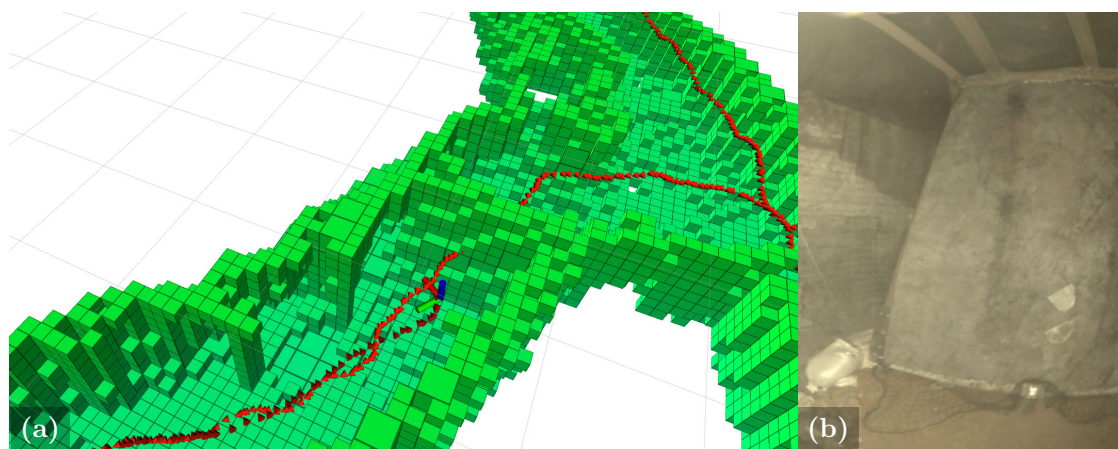


Figure 41. The artificial wall that blocked the way back for UAV *green* in the map (a) and in the camera image (b).



Figure 42. The landing events of all three UAVs. The UAV *red* (a) collided with the Spot UGV, UAV *green* (b) hit a metal rod protruding from the wall, and UAV *blue* (c) landed after its battery was exhausted by hovering while being trapped in a map corrupted by drift in the featureless corridor.



Figure 43. The only three image frames of the survivor *s2* captured by the downward-facing camera. The artifact was not detected as there is only a small part of the survivor's sleeve visible in the image, which is also degraded by motion blur.

Both vent *v1* and drill *d1* were detected, localized, and correctly labeled. The drill *d4* was incorrectly classified as a backpack, nevertheless, the operator reported the correct class based on the detection image. All three UAVs detected the *d4* drill, but UAV *green* provided the highest accuracy, which is reported in [Table 6](#). In total, four artifact hypotheses arrived to the base station with sufficient information for obtaining a point for the report.

Table 5. Statistics of artifact detection for each deployed UAV from the prize round of the Final Event. The *seen* column yields the number of artifacts that appeared in the image of one of the on-board cameras. If the artifact was detected by the CNN, it is listed in the *detected* column and the detection is shown in Figure 45. Artifacts that were *confirmed* had enough consistent detections to establish a hypothesis. *Confirmed unique* artifacts were not detected by another robot, including UGVs.

UAV	Artifacts			
	seen	detected	confirmed	confirmed unique
Red	1	1	1	0
Green	4	3	3	1
Blue	4	4	3	1

Table 6. Unique artifacts detected by lightweight CNN running on-board UAVs in real time. The total error e_{tot} of the artifact position is the sum of the UAV localization drift error e_{loc} and the error of estimating the artifact position e_{est} from the detected bounding box. Artifacts detected by more UAVs are listed only once with values from the most accurate hypothesis among the UAVs. The hypothesis was *Confirmed* when more than four images were associated with it. Some artifacts were correctly detected and localized, but the wrong label was assigned to them. This is documented in the *Correct class* column. Even with a wrong label, the operator could still deduce the correct class by looking at the image sent with the hypothesis. Only one image was sent with each hypothesis, and if it was possible to deduce the correct class, then the image was listed as *Correct image*.

Artifact	Frames detected	Confirmed	Correct class	Correct image	e_{loc} (m)	e_{est} (m)	e_{tot} (m)
v1	27	✓	✓	✓	1.94	4.61	3.08
s1	60	✓	×	×	2.93	4.57	2.89
p1	1	×	×	×	-	-	-
d4	11	✓	×	✓	0.77	1.61	1.30
f1	13	✓	✓	✓	0.85	1.33	1.31
d1	9	✓	✓	✓	1.46	2.30	1.55

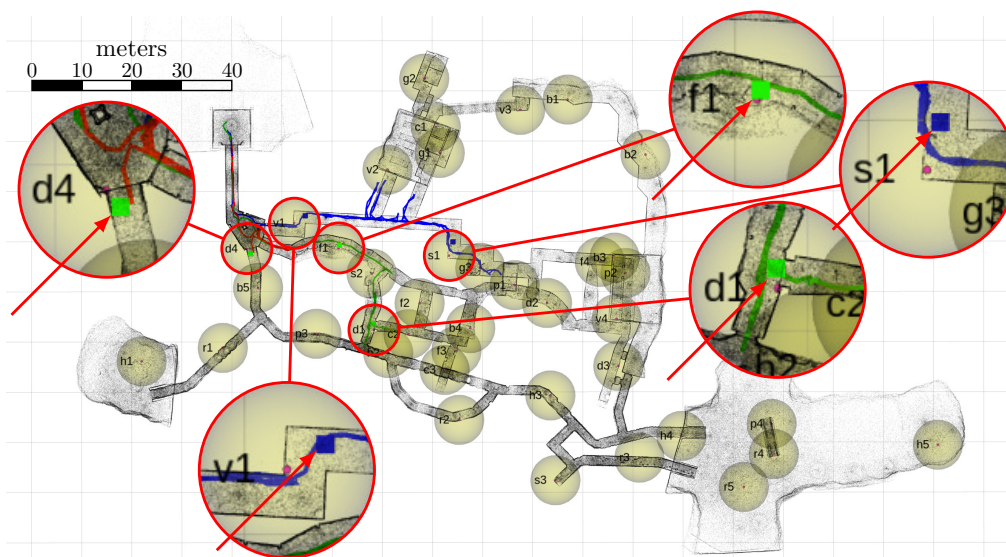


Figure 44. The map of the Final Event course was obtained by the organizers by scanning the course with a laser scanner station. The paths traveled by all three UAVs (*red*, *green*, and *blue*) during the Final Event are depicted by their respective colors. The ground truth positions of artifacts are surrounded by a yellow sphere in order to visualize the 5 m limit for the reported artifact to be counted as a point in the competition. The five artifacts that were detected and localized within this 5 m limit are shown as squares colored by the detecting UAV and highlighted in the magnified sections with red arrows.



Figure 45. Images of artifacts detected by the UAVs in the Final Event. The color of the rectangle shows which UAV detected the artifact and at what mission time as shown in the bottom right corner.

14.3. DARPA SubT Final Event Virtual Track

In parallel to the Systems Track, the competition was also running in the simulated form of the Virtual Track. The teams had to submit a solution consisting of docker images of a robotic team put together within a limited budget to buy the robots and their sensory packages.

The Systems Track included a single run (with two preliminary rounds) conducted in a single world and was therefore focused on the reliability of the robots, which had to overcome challenging terrain with narrow passages and adverse conditions for perception. On the other hand, the virtual teams were deployed three times in each of the eight worlds, ranging from vast models of artificially created environments to scanned courses from the previous events, including the Final Event course. Moreover, in the Virtual Track, the whole mission must be fully autonomous and no human interventions are possible. The purpose of the virtual event was to evaluate the high-level planning, cooperation, decision-making, and efficient coverage of the large worlds. As the cooperative searching strategy is one of the core contributions of this work, we have presented the results from the virtual course here as most of the worlds allowed for efficient deployment and cooperation of the multirobot teams.

14.3.1. Differences from the Systems Track

The simulation model of the IMU provides much better data compared to the real sensor with the same parameters. Thus is due to the measurements in the simulation not being corrupted by propeller-induced vibrations, wind gusts, or saturation, as well as having the IMU rigidly attached to the UAV body with known extrinsic parameters. The higher quality of the simulated data allows for the use of LiDAR-inertial odometry. In addition to the LiDAR, it also relies on the IMU preintegration in its optimization process, thus providing a smooth and drift-free position estimate, even when there are few geometrically rich features present. Specifically, the LIO-SAM (Shan et al., 2020) algorithm was chosen for its low drift and high precision over the A-LOAM deployed in the Systems Track. Both algorithms are detailed in Section 6.

The computation power available for artifact detection in the Virtual Track was not constrained by the weight of the onboard computation hardware as was the case in the Systems Track. As a result, compromises of the performance/weight ratio had to be made on the detector in the Systems Track, as reported in Section 10.

Reporting of the found artifacts is handled by the operator in the Systems Track, which is not possible in the fully autonomous Virtual Track. A virtual artifact reporter algorithm was developed to gather artifact hypotheses from all robots and decide which hypotheses are the most likely to score a point (described in detail in [Section 10.4](#)).

The control interface of the simulated UAV was also different from the real one. While the FCU of the real UAV accepted attitude rate commands generated by the Special Euclidean group of dimension 3 (SE(3)) controller, the simulated UAV was controlled on a higher level by velocity commands. This did not allow for precise control of the UAV motion, as was the case for the low-level attitude rate control.

The deployment sequence of individual robots in the Systems Track could be decided by the operator based on the requirements of locomotion modality, dynamics, and sensory payload during the progress of the mission. In contrast, the sequence in the Virtual Track was fixed before the start of the run.

LandMap introduced in [Section 7.5](#) was not used in the Virtual Track where the UAV was not destroyed even after a rough landing. As long as the UAV landed in the communication range of the network it could send its hypotheses to the base station and further retranslate messages from/to other robots.

14.3.2. Virtual Track results

In the virtual deployment, our team consisted of five UAVs and two UGVs. The UAVs were the superior platform in the Virtual Track due to their greater movement speed, smaller form-factor, and better mobility to fly over terrain untraversable by the UGVs. We deployed two UGVs to build a communication network consisting of breadcrumbs dropped at the edges of the wireless signal range. This allowed for the UAVs to maximize the time for searching for artifacts as they could return to the nearest breadcrumb instead of to the base station back at the staging area. Both UGVs were deployed at the start of the run. The deployment times and exploration strategies of individual UAVs are listed in [Table 7](#). Our solution achieved 2nd place with a total of 215 scored points. [Table 8](#) summarizes the points scored by the top three teams on each world of the Virtual Track ([Figure 46](#)). The lower number of points on worlds 4, 5, 6, and 8 can be explained by the fact that these worlds were not made of the tiles that were used in the qualification and practice worlds. The details on traveled distance and collected hypotheses by particular UAVs during all runs of the SubT Virtual Finals are provided in [Figure 47](#) and [Figure 48](#), respectively.

Table 7. The times of deployment and assigned strategies from [Section 8.2](#) in the Virtual Track. The second UAV was scheduled to take off after the first UAV returned to communication range so that it can take advantage of the LTVMap of the first UAV. Both DEI and Greedy strategies were used as DEI guarantees covering dead-end corridors at the cost of lower average velocity and lower total surface covered. The first UAV used DEI so that the rest of the team did not need to return to where the first UAV already had been. The next two UAVs maximize the searched volume with the Greedy strategy and the last two UAVs cover any missed surfaces with DEI.

UAV	1	2	3	4	5
Start (s)	60	1560	1680	1800	1920
Strategy	DEI	Greedy	Greedy	DEI	DEI

Table 8. The score achieved by the top three teams on each world of the Virtual Track. The reported values are the sums of three runs on each world.

World	1	2	3	4	5	6	7	8	total
Dynamo	21	52	48	18	15	11	44	14	223
CTU-CRAS-NORLAB	31	39	45	16	18	13	36	17	215
Coordinated Robotics	44	41	27	23	17	14	26	20	212

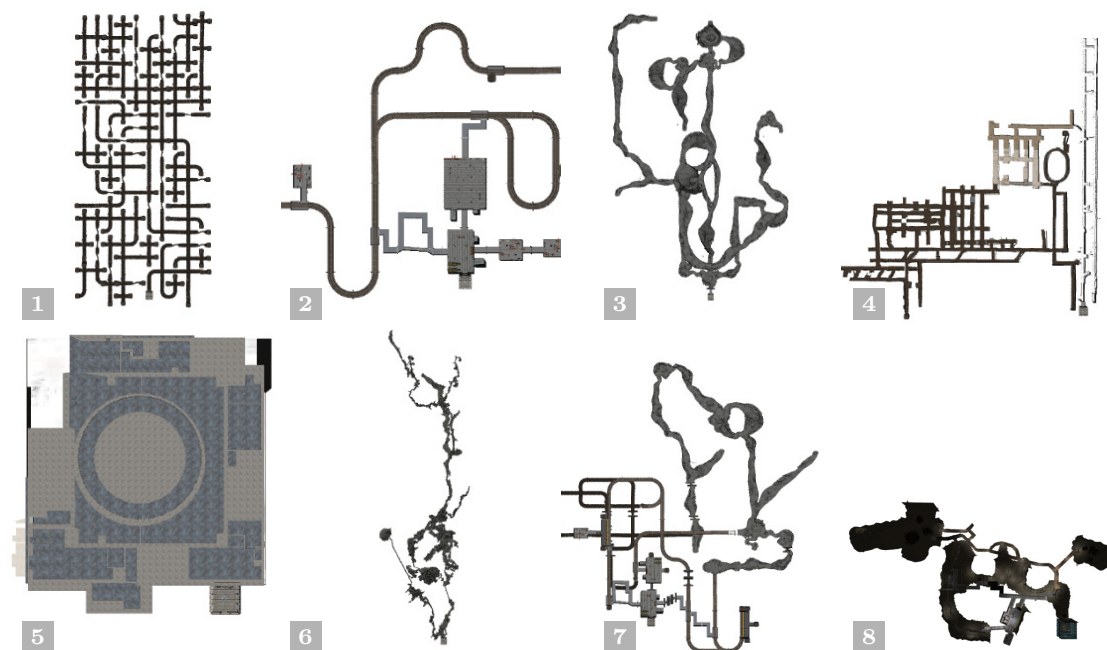


Figure 46. All eight worlds used in the Virtual Track of the DARPA SubT Finals. The worlds 1, 2, 3, and 7 are built from tiles that were used in the preliminary and practice rounds. World 4 is the model of the NIOSH research mine, where the tunnel circuit was held. Similarly, world 5 corresponds to the model of the location of the urban circuit—the unfinished Satsop nuclear power plant. World 6 is a model of a narrow cave system. World 8 is modeled based on the Systems Track Finals.

15. Lessons learned and future work

In this section, we present our view on the state of the S&R UAVs, the lessons learned, which problems are solved, and what areas require more research to achieve reliable performance suitable for deployment as a tool for assisting rescue workers. These findings were collected throughout the preparation for as well as during the DARPA SubT Competition, which aimed to push the state of the art of S&R robotics. Furthermore, this discussion should be of some interest to the community as we highlight aspects that could be explored in future research and development. In general, most of the individual subproblems, such as localization, mapping, detection, and communication, are solved to the point of being capable of performing an autonomous mission in extremely challenging conditions. The developed algorithms are now used in actual field deployment instead of just laboratories and simulations, which introduces disturbances, noise, dust, and other detrimental effects that negatively impact the algorithms' performance and reliability. It is essential to focus on the reliability of the employed methods to make the UAVs a valuable asset to the S&R team.

The role of the aerial robot in a heterogeneous S&R robotic team is a quickly deployable agent that can provide swift situation awareness, environment type, and topology information that allows for informed decision-making about the rest of the mission. Furthermore, areas such as caves, collapsed buildings or high openings can often be reached only by UAVs. On the other hand, ground robots have the advantage of higher payload capacity, which results in improved perception capabilities compared to UAVs.

The localization method based on 3D LiDAR provides precise position estimates, even under severe degradation by dust. However, as proved by the UAV *blue*, the estimate can begin to drift when the solved optimization is ill-conditioned due to low-variance geometry, typically in long corridors with straight walls. The unpredictable nature of subterranean environments requires a localization method that is reliable and drift-free under arbitrary conditions. Solutions based

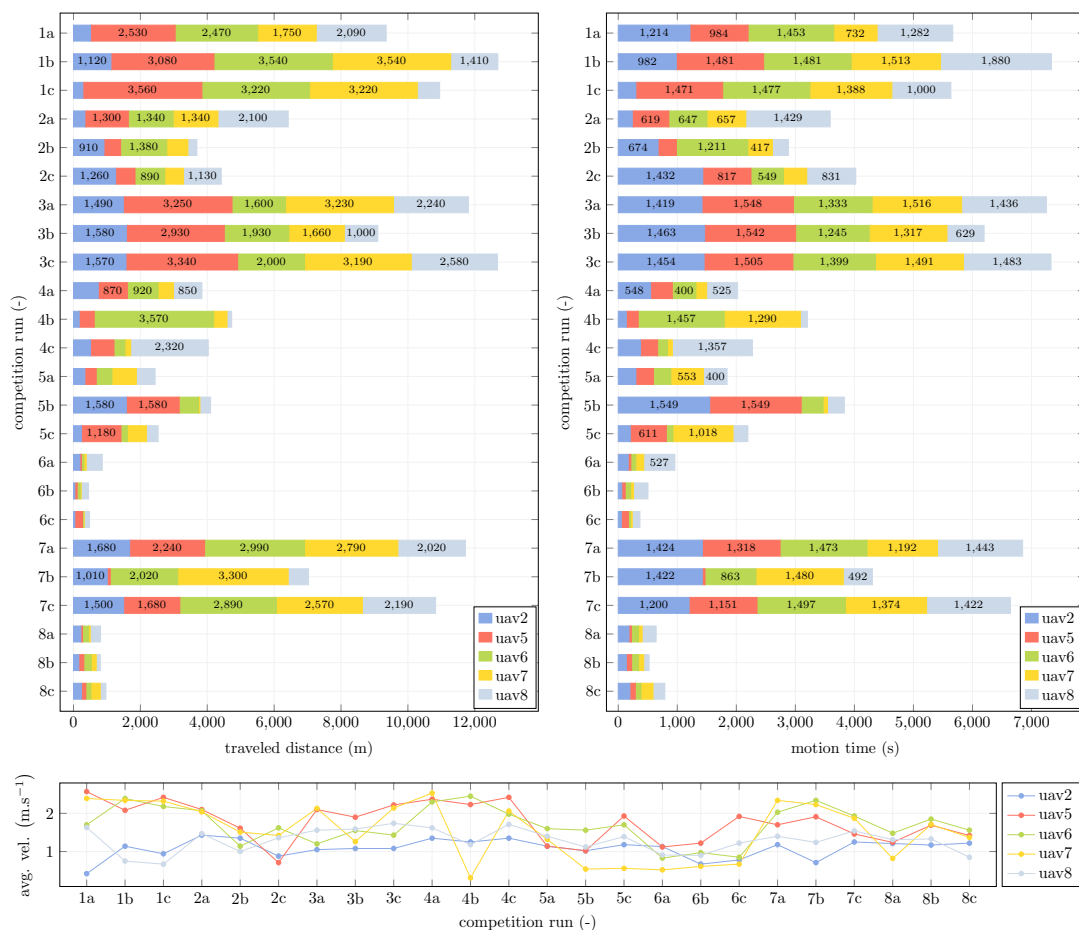
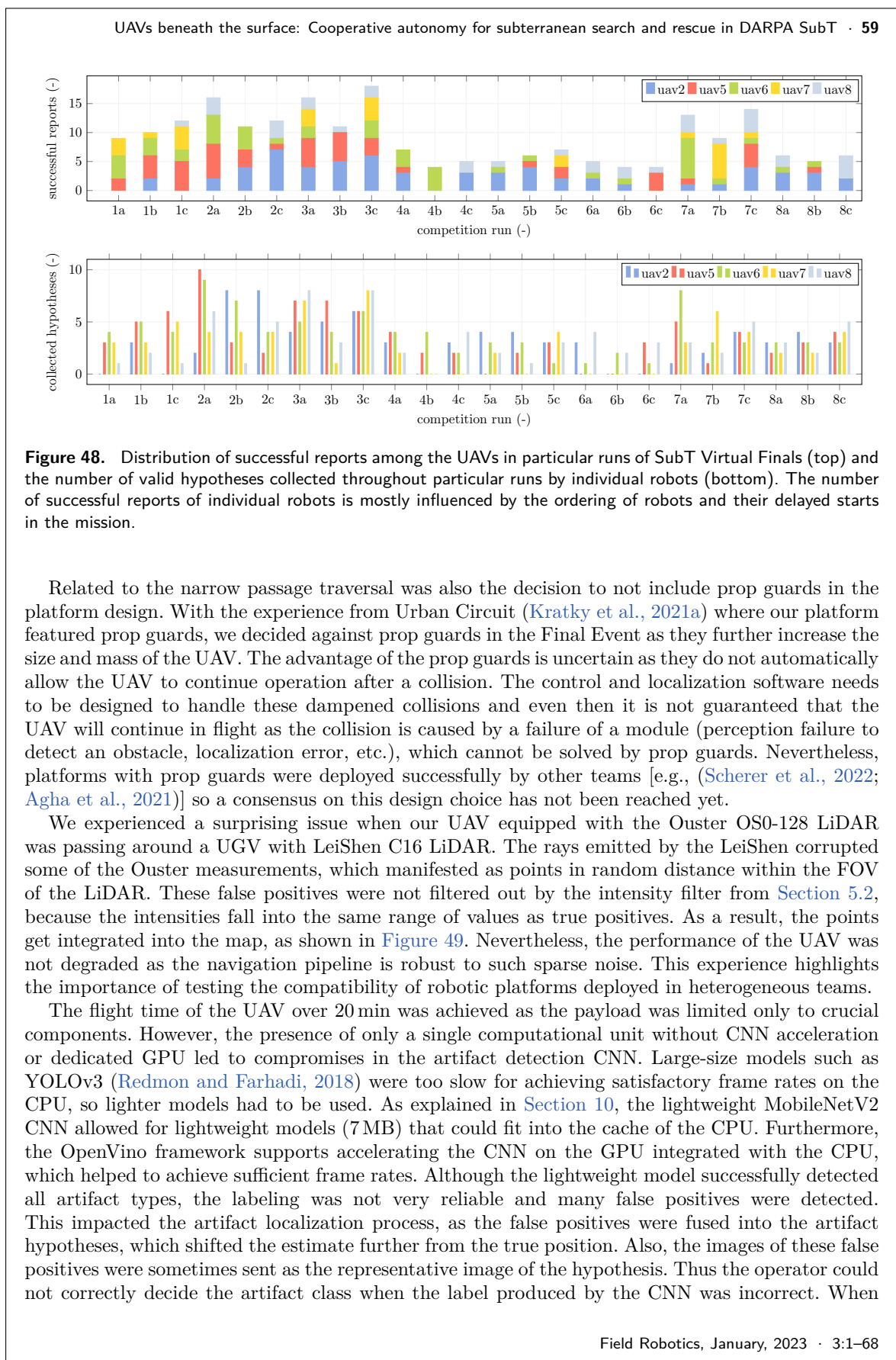


Figure 47. Overall traveled distance, time of active motion, and average velocity of particular UAVs in all runs of the SubT Virtual Finals. The maximum traveled distance throughout all runs was achieved by UAV5 in run 1c (3560 m). The maximum active time was achieved by UAV2 in run 5b (1539 s). The presented average velocity incorporates the entire flight, including hovering states.

on detecting geometrical degeneracy, and multimodal fusion of LiDAR and visual methods were described in Section 2.2. The results seem promising but due to high unpredictability and challenges of subterranean environments more research in localization algorithms is still required for truly robust pose estimation in arbitrary conditions.

In addition to map drift caused by errors in the localization, the volumetric occupancy grid did not contain the smaller obstacles like ropes, cables, and thin poles, which led to the collision of UAV *green* as seen in Figure 42b. Although some LiDAR rays hit these thin obstacles, the occupied cells generated by these rays were often changed to free when multiple rays that passed through these cells hit the wall behind them. As a result, the navigation pipeline planned a path through these cells that appeared free, but contained a thin metal pole, causing a collision. The ability to traverse narrow passages is also impaired since the passages appear narrower than they really are due to grid discretization. We propose to locally increase the resolution of the grid of DenseMap on demand to approximate the free space more accurately, while keeping the scan integration times bounded. This approach is however only a partial solution as the need for a more granular resolution might not always be reliably detected. Consequently, the need arises for a flexible map that is not bound by fixed cell size, similarly to the SphereMap, possibly based on surfel mapping as seen in (Behley and Stachniss, 2018), or based on GMM (O’Meadhra et al., 2018).



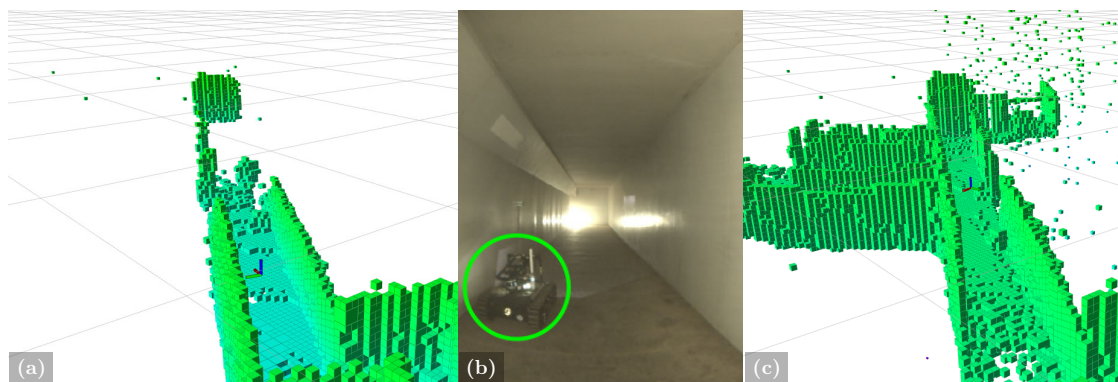


Figure 49. DenseMap before (a) approaching the UGV with LeiShen C16 LiDAR (b) and (c) when it gets corrupted by random points in the FOV of the LiDAR mounted on the UAV after flying in close vicinity (≈ 1 m) to the UGV. Notice, a few false positives were integrated into the map even when the UAV was 8 m away from the UGV (a).

payload capacity prevents the use of more capable hardware, the issue must be compensated by the sensing strategy. In contrast to UGVs the mobility of UAVs allows reaching closer to the artifact to verify the detection. Approaches of perception-driven navigation can improve the performance of the lightweight detector by planning a trajectory to inspect the artifact from a closer distance and other angles after the initial detection.

Although our platform is quite compact (500 mm without propellers), it could not pass through all of the narrow passages, even during the postevent testing. Apart from the discrete map and conservatively set distance from obstacles (see Table 4), the size of the UAV prevented flying through some of the narrower passages of the circuit. As even smaller passages are to be expected during deployment of robots in real S&R scenarios, the UAV platforms should be further miniaturized to allow safe traversal of narrow passages. Deployment of visually localized UAVs could decrease the size significantly but the capabilities of visual navigation pipelines still underperform compared to the LiDAR solutions, which was the preferred approach of most teams. A possible workaround that compensates for the lower flight time of smaller platforms is the marsupial deployment (Lindqvist et al., 2022; De Petris et al., 2022). When such miniaturization is not possible due to, e.g., insufficient payload of smaller platforms, a heterogeneous aerial team consisting of both large and small platforms can be deployed. In such case, the large platform carrying a LiDAR can command and send position corrections to smaller visually localized UAV that can inspect tight narrow passages that are unreachable by the large UAV (Pritzl et al., 2022b).

A mutual collision avoidance module is a necessity for any application where multiple robots share the same workspace. The developed priority-based module uses the already shared information about the robots' positions when communication is available, as it should since the risk of collision arises when robots are in close proximity. This module prevented collisions in the Virtual Track, where despite the vastness of most of the worlds, the collisions happened often in the practice runs before implementing the collision avoidance. We decided against using the collision avoidance module in the Systems Track. This was done as the robots could easily become deadlocked in tight corridors and also due to the collision probability being reasonably low because of the delay between each UAVs launch. Additionally, the operator could override the full autonomy to prevent collision, if necessary. Nevertheless, the UAV *red* collided with a Spot UGV shortly after the start of the run, which could have been prevented if collision avoidance was enabled. A deadlock-free solution based on agent theory approaches can be devised for situations when communication is available, and behavior prediction methods can provide a backup when communication is not possible.

Even though the organizers did a great job at providing a realistic simulation environment for the Virtual Track, many phenomena, unexpected situations, and issues from the real world are

not present in the simulation. Moreover, the rules of the competition are different for the two tracks. For example, the absence of a human operator in the virtual team changes the approach drastically as all decision-making needs to be automated. For details about the differences between the Systems and Virtual tracks see [Section 14.3.1](#). As a team that participated in both Systems and Virtual tracks, we want to list the greatest hurdles encountered in the simulation to real-world transfer:

- **Degraded sensor output.** The output of real-world sensors is corrupted by various negative effects. On the other hand, the imperfections in simulated sensors are typically modeled only by an additive noise, which most algorithms can cope with by smoothing or filtering. The performance of such algorithms severely deteriorates with input data degraded by the harsh conditions of underground environments. In the case of laser-based sensors, rays get reflected by small airborne particles such as dust, smoke, and fog to produce false measurements. Although the fog was modeled in the simulation, the distribution of fog points in the point cloud did not coincide with the distribution in the real world, and thus we implemented different approaches for Virtual and Systems tracks, which are detailed in [Section 5.2](#). Cameras in addition to the airborne particles suffer from insufficient illumination, high-contrast scenes, motion blur, and rolling shutter aliasing. Using neural networks for object detection proved to be robust to these effects when trained on datasets with similar data.
- **Environment scale.** The scale of the environment was much smaller in the Systems Track than in the Virtual Track. Most of the simulation worlds spanned several-kilometers-long corridors with vast caves to challenge the cooperative exploration abilities of the teams. The vastness and openness of the worlds favored fast flights to cover as much space as possible during flight time. In contrast, the Systems Track was narrow from the beginning of the course (see the cross-section distribution in [Table 2](#)) and the UAV was closer than 1 m from obstacles for most of the flight (see the distance to obstacle plot in [Figure 40](#)). To minimize the chance of collision, the velocity of the UAV was constrained to 1 m s^{-1} instead of 5 m s^{-1} in simulation and the control was tuned for low error by commanding the UAV in attitude rates instead of linear velocities.
- **Safety.** In simulation, the robots cannot harm anyone and to perform another run after a collision it is sufficient to restart the simulation. Contrary to that, in the real world, special care must be taken to make the robots, especially fast aerial robots with quickly spinning propellers, safe for the environment, operators, and any other humans in the vicinity. To assure maximum safety during takeoff, status checks are performed automatically but then the safety operator has to approve the takeoff by toggling a switch on the RC as described in [Section 11.1](#). During the flight, health checks of the rate of crucial data streams are performed, control errors are monitored, and innovation of state estimation corrections are analyzed. When any of the monitored values exceeds a critical threshold, an emergency landing is initiated to minimize the damage to the platform. A technique based on data from the downward-facing depth camera was developed to assure safe regular (not emergency) landing on planar low-slope surfaces stored in LandMap ([Section 7.5](#)).
- **Decision-making in artifact reporting.** Due to the limited payload of the UAV, a lightweight CNN was used for artifact detection, which forced us to choose a compromise between precision and recall. Having a human operator to verify the artifact hypotheses allowed us to maximize recall to not miss any artifact at the cost of a higher false positive count. A correctly detected but misclassified artifact could still score a point if the operator correctly deduced the class from the image (see [Table 6](#)). This would not have been possible using the autonomous arbiter and thus the flexibility of a human operator overperforms the autonomous arbiter, which optimizes a fixed criterion when reporting an artifact hypothesis.
- **Computational resources** The simulated run in Virtual Track was running only at a fraction of real time and thus the employed algorithms had more computation time available. In the Systems Track, the developed algorithms had to run on the onboard processing units in real

time. Thus, for the simulation to real-world transfer, the algorithms need to be optimized, critical systems prioritized, and often a compromise between accuracy and computation time has to be found. We discuss all modifications and optimizations in [Section 13](#).

16. Conclusion

This paper has presented the complex UAV system deployed in the final round of the DARPA SubT Challenge after 3 years of development and testing in numerous real world demanding environments (including gold mine, coal mine, abandoned nuclear power plant, caverns, military fortress, natural caves, old factory hall, subway station, etc.). Based on these unique opportunities and experience, we have designed both the hardware UAV platform and the multi-UAV software with a focus on the exploration of such vast, complicated, and varying environments.

In the Systems Track of DARPA SubT Challenge, three UAVs were deployed alongside ground robots into the competition course consisting of a heterogeneous environment of the tunnel, urban, and cave sections, where the aerial team detected and localized four artifacts and traveled 492 m in total. The austere conditions of the circuit, such as narrow passages, dust, featureless corridors, and dynamic obstacles, tested the reliability of the system as a whole, including the hardware design of a compact platform with a considerable flight time of 25 min. Most of the testing was realized in environments where it performed exceptionally well, including a former brewery where the UAV had to explore an abandoned building with partially collapsed floor and ceiling, or during the exploration of Byci Skala (Bull Rock Cave) in the Moravian Karst cavern system. Compared to ground robots the UAVs could search a larger volume of space because they could easily fly over any encountered problematic terrain such as mud, water, and rubble and thus had an advantage in the exploration of unknown terrains with unexpected obstacles. Furthermore, the worlds of the Virtual Track of the competition were also very large; even with our UAV possessing a 25 min flight time and fast dynamics, they were not able to reach the furthest parts of some worlds. Although our system was designed primarily for these large-scale environments, its performance in the challengingly tight corridors of the prize round was also impressive. The difficulty of UAV deployment in such adverse environments motivated numerous achievements beyond the state of the art that are summarized in this paper. Many lessons were learned in the process that could facilitate and support designing complex robotic systems in similar applications in the future.

A larger team of five aerial robots was deployed in the Virtual Track, alongside two UGVs. By employing the proposed cooperative exploration strategies based on topological map sharing, the exploration effort of our team was spread out over a wider area. Together with dynamic flight and reliable artifact detection/localization, this helped to achieve the 2nd place with 215 scored points. Moreover, seven out of the nine participating teams used our X500 UAV, which was modeled according to the specification of the physical platform thanks to its long flight time, a wide array of sensors, modest size, and reasonable price.

Based on the successful deployment in the DARPA SubT, which focused on providing challenging conditions typically encountered during rescue missions in underground environments, we conclude that the presented UAV system is a valuable addition to teams of first responders, as it can provide situational awareness and even find survivors after a catastrophe without risking the lives of rescuers in dangerous environments.












Acknowledgments

We would like to thank the members of the CTU-CRAS-NORLAB team who participated in the design and development of the UAV and UGV hardware platforms, software development, simulations, testing and general support. Namely: Ruslan Agishev, Afzal Ahmad, Teymur Azayev, Jan Bayer, Tommy Bouchard-Lebrun, Petr Čížek, Simon-Pierre Deschênes, Jan Faigl, Olivier Gamache, Alexandre Guénette, Bedřich Himmel, Jakub Janoušek, Tomáš Krajník, Vladimír Kubelka, Denis Ouellet, Tomáš Petříček, François Pomerleau, Miloš Prágr, Tomáš Rouček, Vojtěch Šalanský, Martin

Škarytka, Vojtěch Spurný, Pavel Stoudek, Arsen Tkachev, Maxime Vaidis, Volodymyr Zabulskyi, Karel Zimmermann, and Martin Zoula.

This work was partially funded by the Defense Advanced Research Projects Agency (DARPA), by the CTU grant no. SGS20/174/OHK3/3T/13, by the Czech Science Foundation (GACR) under research project no. 20-29531S, by TAČR project no. FW03010020, by the OP VVV funded project CZ.02.1.01/0.0/0.0/16 019/0000765 “Research Center for Informatics,” by the European Union’s Horizon 2020 research and innovation programme AERIAL-CORE under grant agreement no. 871479, and by the NAKI II project no. DG18P02OVV069.

ORCID

Matěj Petrlik  <https://orcid.org/0000-0002-5337-9558>
 Pavel Petráček  <https://orcid.org/0000-0002-0887-9430>
 Vít Krátký  <https://orcid.org/0000-0002-1914-742X>
 Tomáš Musil  <https://orcid.org/0000-0002-9421-6544>
 Yurii Stasinchuk  <https://orcid.org/0000-0002-2197-1442>
 Matouš Vrba  <https://orcid.org/0000-0002-4823-8291>
 Tomáš Báča  <https://orcid.org/0000-0001-9649-8277>
 Daniel Heřt  <https://orcid.org/0000-0003-1637-6806>
 Martin Pecka  <https://orcid.org/0000-0002-0815-304X>
 Tomáš Svoboda  <https://orcid.org/0000-0002-7184-1785>
 Martin Saska  <https://orcid.org/0000-0001-7106-3816>

References

- Agha, A., Otsu, K., Morrell, B., Fan, D. D., Thakker, R., Santamaria-Navarro, A., Kim, S., Bouman, A., Lei, X., Edlund, J. A., Ginting, M. F., Ebadi, K., Anderson, M., Pailevanian, T., Terry, E., Wolf, M. T., Tagliabue, A., Vaquero, T. S., Palieri, M., et al. (2021). Nebula: Quest for robotic autonomy in challenging environments; TEAM costar at the DARPA subterranean challenge. *CoRR*, abs/2103.11470.
- Ahmad, A., Walter, V., Petracek, P., Petrlik, M., Baca, T., Zaitlik, D., and Saska, M. (2021). Autonomous aerial swarming in gnss-denied environments with high obstacle density. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 570–576. IEEE.
- Ahmad, S., Sunberg, Z. N., and Humbert, J. S. (2021). End-to-end probabilistic depth perception and 3d obstacle avoidance using pomdp. *Journal of Intelligent & Robotic Systems*, 103(2):1–18.
- Alismail, H., Kaess, M., Browning, B., and Lucey, S. (2016). Direct visual odometry in low light using binary descriptors. *IEEE Robotics and Automation Letters*, 2(2):444–451.
- Alotaibi, E. T., Alqefari, S. S., and Koubaa, A. (2019). Lsar: Multi-uav collaboration for search and rescue missions. *IEEE Access*, 7:55817–55832.
- Baca, T., Hert, D., Loiano, G., Saska, M., and Kumar, V. (2018). Model Predictive Trajectory Tracking and Collision Avoidance for Reliable Outdoor Deployment of Unmanned Aerial Vehicles. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1–8. IEEE.
- Baca, T., Loiano, G., and Saska, M. (2016). Embedded Model Predictive Control of Unmanned Micro Aerial Vehicles. In *2016 IEEE International Conference on Methods and Models in Automation and Robotics (MMAR)*, pages 992–997.
- Baca, T., Petrlik, M., Vrba, M., Spurny, V., Penicka, R., Hert, D., and Saska, M. (2021). The mrs uav system: Pushing the frontiers of reproducible research, real-world deployment, and education with autonomous unmanned aerial vehicles. *Journal of Intelligent & Robotic Systems*, 102(26):1–28.
- Bayer, J. and Faigl, J. (2020). Handheld localization device for indoor environments. In *2020 4th International Conference on Automation, Control and Robots (ICACR)*, pages 60–64.
- Behley, J. and Stachniss, C. (2018). Efficient surfel-based slam using 3d laser range data in urban environments. In *Robotics: Science and Systems*, volume 2018, page 59.
- Bircher, A., Kamel, M., Alexis, K., Oleynikova, H., and Siegwart, R. (2016). Receding horizon” next-best-view” planner for 3d exploration. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 1462–1468. IEEE.

- Blochlinger, F., Fehr, M., Dymczyk, M., Schneider, T., and Siegwart, R. Y. (2018). Topomap: Topological mapping and navigation based on visual slam maps. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–9.
- Bosse, M., Zlot, R., and Flick, P. (2012). Zebedee: Design of a spring-mounted 3-d range sensor with application to mobile mapping. *IEEE Transactions on Robotics*, 28(5):1104–1119.
- Burgard, W., Moors, M., Stachniss, C., and Schneider, F. (2005). Coordinated multi-robot exploration. *Robotics, IEEE Transactions on*, 21:376 – 386.
- Burri, M., Oleynikova, H., , Achtelik, M. W., and Siegwart, R. (2015). Real-Time Visual-Inertial Mapping, Re-localization and Planning Onboard MAVs in Unknown Environments. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems*.
- Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I., and Leonard, J. J. (2016). Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6):1309–1332.
- Chang, Y., Ebadi, K., Denniston, C. E., Ginting, M. F., Rosinol, A., Reinke, A., Palieri, M., Shi, J., Chatterjee, A., Morrell, B., Agha-mohammadi, A.-a., and Carlone, L. (2022). Lamp 2.0: A robust multi-robot slam system for operation in challenging large-scale underground environments. *IEEE Robotics and Automation Letters*, 7(4):9175–9182.
- Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., Zhang, Z., Cheng, D., Zhu, C., Cheng, T., Zhao, Q., Li, B., Lu, X., Zhu, R., Wu, Y., Dai, J., Wang, J., Shi, J., Ouyang, W., Loy, C. C., and Lin, D. (2019). MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*.
- Chen-Lung, L., Jui-Te, H., Huang, C.-I., Zi-Yan, L., Chao-Chun, H., Yu-Yen, H., Siao-Cing, H., Po-Kai, C., Zu Lin, E., Po-Jui, H., Po-Lin, L., Bo-Hui, W., Lai-Sum, Y., Sheng-Wei, H., MingSian R., B., and Hsueh-Cheng, W. (2022). A heterogeneous unmanned ground vehicle and blimp robot team for search and rescue using data-driven autonomy and communication-aware navigation. *Field Robotics*, 2:557–594.
- Dang, T., Khattak, S., Mascarich, F., and Alexis, K. (2019a). Explore locally, plan globally: A path planning framework for autonomous robotic exploration in subterranean environments. In *2019 19th International Conference on Advanced Robotics (ICAR)*, pages 9–16. IEEE.
- Dang, T., Mascarich, F., Khattak, S., Nguyen, H., Nguyen, H., Hirsh, S., Reinhart, R., Papachristos, C., and Alexis, K. (2020a). Autonomous search for underground mine rescue using aerial robots. In *2020 IEEE Aerospace Conference*, pages 1–8. IEEE.
- Dang, T., Mascarich, F., Khattak, S., Papachristos, C., and Alexis, K. (2019b). Graph-based path planning for autonomous robotic exploration in subterranean environments. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3105–3112.
- Dang, T., Tranzatto, M., Khattak, S., Mascarich, F., Alexis, K., and Hutter, M. (2020b). Graph-based subterranean exploration path planning using aerial and legged robots. *Journal of Field Robotics*, 37(8):1363–1388.
- De Petris, P., Khattak, S., Dharmadhikari, M., Waibel, G., Nguyen, H., Montenegro, M., Khedekar, N., Alexis, K., and Hutter, M. (2022). Marsupial walking-and-flying robotic deployment for collaborative exploration of unknown environments. *arXiv preprint arXiv:2205.05477*.
- Delmerico, J., Mintchev, S., Giusti, A., Gromov, B., Melo, K., Horvat, T., Cadena, C., Hutter, M., Ijspeert, A., Floreano, D., et al. (2019). The current state and future outlook of rescue robotics. *Journal of Field Robotics*, 36(7):1171–1191.
- Denniston, C. E., Chang, Y., Reinke, A., Ebadi, K., Sukhatme, G. S., Carlone, L., Morrell, B., and Agha-mohammadi, A.-a. (2022). Loop closure prioritization for efficient and scalable multi-robot slam. *IEEE Robotics and Automation Letters*, 7(4):9651–9658.
- Ebadi, K., Bernreiter, L., Biggie, H., Catt, G., Chang, Y., Chatterjee, A., Denniston, C. E., Deschenes, S.-P., Harlow, K., Khattak, S., et al. (2022). Present and future of slam in extreme underground environments. *arXiv preprint arXiv:2208.01787*.
- Ebadi, K., Chang, Y., Palieri, M., Stephens, A., Hatteland, A., Heiden, E., Thakur, A., Funabiki, N., Morrell, B., Wood, S., et al. (2020). Lamp: Large-scale autonomous mapping and positioning for exploration of perceptually degraded subterranean environments. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 80–86. IEEE.
- Ebadi, K., Palieri, M., Wood, S., Padgett, C., and Agha-mohammadi, A.-a. (2021). Dare-slam: Degeneracy-aware and resilient loop closing in perceptually degraded environments. *Journal of Intelligent & Robotic Systems*, 102(1):1–25.

- Fabris, A., Kirchgeorg, S., and Mintchev, S. (2021). A soft drone with multi-modal mobility for the exploration of confined spaces. In *2021 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 48–54.
- Fan, D. D., Otsu, K., Kubo, Y., Dixit, A., Burdick, J., and Agha-Mohammadi, A.-A. (2021). Step: Stochastic traversability evaluation and planning for safe off-road navigation. *arXiv preprint arXiv:2103.02828*.
- Ginting, M. F., Otsu, K., Edlund, J. A., Gao, J., and Agha-Mohammadi, A.-A. (2021). Chord: Distributed data-sharing via hybrid ros 1 and 2 for multi-robot exploration of large-scale complex environments. *IEEE Robotics and Automation Letters*, 6(3):5064–5071.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in neural information processing systems*, 27.
- Harabor, D. and Grastien, A. (2011). Online graph pruning for pathfinding on grid maps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 25, pages 1114–1119.
- Hert, D., Baca, T., Petracek, P., Kratky, V., Spurny, V., Petrlik, M., Matous, V., Zaitlik, D., Stoudek, P., Walter, V., Stepan, P., Horyna, J., Pritzl, V., Silano, G., Bonilla Licea, D., Stibinger, P., Penicka, R., Nascimento, T., and Saska, M. (2022). MRS Modular UAV Hardware Platforms for Supporting Research in Real-World Outdoor and Indoor Environments. In *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE.
- Hess, W., Kohler, D., Rapp, H., and Andor, D. (2016). Real-time loop closure in 2d lidar slam. In *2016 IEEE international conference on robotics and automation (ICRA)*, pages 1271–1278. IEEE.
- Hornung, A., Wurm, K. M., Bennewitz, M., Stachniss, C., and Burgard, W. (2013). OctoMap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees. *Autonomous Robots*, 34:189–206.
- Huang, Y.-W., Lu, C.-L., Chen, K.-L., Ser, P.-S., Huang, J.-T., Shen, Y.-C., Chen, P.-W., Chang, P.-K., Lee, S.-C., and Wang, H.-C. (2019). Duckiefloat: a collision-tolerant resource-constrained blimp for long-term autonomy in subterranean environments. *arXiv preprint arXiv:1910.14275*.
- Hudson, N., Talbot, F., Cox, M., Williams, J. L., Hines, T., Pitt, A., Wood, B., Frousheger, D., Surdo, K. L., Molnar, T., Steindl, R., Wildie, M., Sa, I., Kottege, N., Stepanas, K., Hernandez, E., Catt, G., Docherty, W., Tidd, B., Tam, B., Murrell, S., Bessell, M., Hanson, L., et al. (2022). Heterogeneous ground and air platforms, homogeneous sensing: Team CSIRO data61’s approach to the DARPA subterranean challenge. *Field Robotics*, 2:595–636.
- Kasper, M., McGuire, S., and Heckman, C. (2019). A benchmark for visual-inertial odometry systems employing onboard illumination. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5256–5263. IEEE.
- Khattak, S., Mascarich, F., Dang, T., Papachristos, C., and Alexis, K. (2019). Robust thermal-inertial localization for aerial robots: A case for direct methods. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1061–1068. IEEE.
- Khattak, S., Nguyen, H., Mascarich, F., Dang, T., and Alexis, K. (2020). Complementary multi-modal sensor fusion for resilient robot pose estimation in subterranean environments. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1024–1029.
- Kohlbrecher, S., Meyer, J., von Stryk, O., and Klingauf, U. (2011). A flexible and scalable slam system with full 3d motion estimation. In *Proc. IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*. IEEE.
- Koval, A., Karlsson, S., Mansouri, S. S., Kanellakis, C., Tevetzidis, I., Haluska, J., Agha-mohammadi, A.-a., and Nikolakopoulos, G. (2022). Dataset collection from a subt environment. *Robotics and Autonomous Systems*, 155:104168.
- Kramer, A., Kasper, M., and Heckman, C. (2021). Vi-slam for subterranean environments. In *Field and Service Robotics*, pages 159–172. Springer.
- Kratky, V., Alcantara, A., Capitan, J., Stepan, P., Saska, M., and Ollero, A. (2021). Autonomous aerial filming with distributed lighting by a team of unmanned aerial vehicles. *IEEE Robotics and Automation Letters*, 6(4):7580–7587.
- Kratky, V., Petracek, P., Baca, T., and Saska, M. (2021a). An autonomous unmanned aerial vehicle system for fast exploration of large complex indoor environments. *Journal of Field Robotics*, 38(8):1036–1058.
- Kratky, V., Petracek, P., Nascimento, T., Cadilova, M., Skobrtal, M., Stoudek, P., and Saska, M. (2021b). Safe documentation of historical monuments by an autonomous unmanned aerial vehicle. *ISPRS International Journal of Geo-Information*, 10(11):738/1–16.
- Kulkarni, M., Dharmadhikari, M., Tranzatto, M., Zimmermann, S., Reijgwart, V., De Petris, P., Nguyen, H., Khedekar, N., Papachristos, C., Ott, L., et al. (2021). Autonomous teamed exploration of subterranean environments using legged and aerial robots. *arXiv preprint arXiv:2111.06482*.

- Lajoie, P.-Y., Ramtoula, B., Chang, Y., Carlone, L., and Beltrame, G. (2020). Door-slam: Distributed, online, and outlier resilient slam for robotic teams. *IEEE Robotics and Automation Letters*, 5(2):1656–1663.
- Lee, T. et al. (2010). Geometric tracking control of a quadrotor UAV on SE(3). In *2010 IEEE Conference on Decision and Control*, pages 5420–5425. IEEE.
- Lei, X., Kim, T., Marchal, N., Pastor, D., Ridge, B., Scholler, F., Terry, E., Chavez, F., Touma, T., Otsu, K., et al. (2022). Early recall, late precision: Multi-robot semantic object mapping under operational constraints in perceptually degraded environments. *arXiv preprint arXiv:2206.10062*.
- Lindqvist, B., Kanellakis, C., Mansouri, S. S., Agha-mohammadi, A.-a., and Nikolakopoulos, G. (2021). Compra: A compact reactive autonomy framework for subterranean mav based search-and-rescue operations. *arXiv preprint arXiv:2108.13105*.
- Lindqvist, B., Karlsson, S., Koval, A., Tevetzidis, I., Haluska, J., Kanellakis, C., Agha-mohammadi, A.-a., and Nikolakopoulos, G. (2022). Multimodality robotic systems: Integrated combined legged-aerial mobility for subterranean search-and-rescue. *Robotics and Autonomous Systems*, 154:104134.
- Loshchilov, I. and Hutter, F. (2016). Sgdr: Stochastic gradient descent with warm restarts.
- Lu, C. X., Rosa, S., Zhao, P., Wang, B., Chen, C., Stankovic, J. A., Trigoni, N., and Markham, A. (2020). See through smoke: robust indoor mapping with low-cost mmwave radar. In *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, pages 14–27.
- Martinez-Rozas, S., Rey, R., Alejo, D., Acedo, D., Cobano, J. A., Rodriguez-Ramos, A., Campoy, P., Merino, L., and Caballero, F. (2022). An aerial/ground robot team for autonomous firefighting in urban gnss-denied scenarios. *Field Robotics*, 2:241–273.
- Moniruzzaman, M., Rassau, A., Chai, D., and Islam, S. M. S. (2022). Teleoperation methods and enhancement techniques for mobile robots: A comprehensive survey. *Robotics and Autonomous Systems*, 150:103973.
- Murphy, R. R., Kravitz, J., Stover, S. L., and Shoureshi, R. (2009). Mobile robots in mine rescue and recovery. *IEEE Robotics & Automation Magazine*, 16(2):91–103.
- Museth, K. (2013). Vdb: High-resolution sparse volumes with dynamic topology. *ACM transactions on graphics (TOG)*, 32(3):1–22.
- Musil, T., Petrlík, M., and Saska, M. (2022). Spheremap: Dynamic multi-layer graph structure for rapid safety-aware uav planning. *IEEE Robotics and Automation Letters*, 7(4):11007–11014.
- Ohradzensky, M. T., Rush, E. R., Riley, D. G., Mills, A. B., Ahmad, S., McGuire, S., Biggie, H., Harlow, K., Miles, M. J., Frew, E. W., et al. (2021). Multi-agent autonomy: Advancements and challenges in subterranean exploration. *arXiv preprint arXiv:2110.04390*.
- Orekhov, V. and Chung, T. (2022). The darpa subterranean challenge: A synopsis of the circuits stage. *Field Robotics*, 2:735–747.
- O’Meadhra, C., Tabib, W., and Michael, N. (2018). Variable resolution occupancy mapping using gaussian mixture models. *IEEE Robotics and Automation Letters*, 4(2):2015–2022.
- Palieri, M., Morrell, B., Thakur, A., Ebadi, K., Nash, J., Chatterjee, A., Kanellakis, C., Carlone, L., Guaragnella, C., and Agha-mohammadi, A.-a. (2020). Locus: A multi-sensor lidar-centric solution for high-precision odometry and 3d mapping in real-time. *IEEE Robotics and Automation Letters*, 6(2):421–428.
- Papachristos, C., Khattak, S., and Alexis, K. (2017). Uncertainty-aware receding horizon exploration and mapping using aerial robots. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 4568–4575. IEEE.
- Papachristos, C., Khattak, S., Mascarich, F., and Alexis, K. (2019a). Autonomous navigation and mapping in underground mines using aerial robots. In *2019 IEEE Aerospace Conference*, pages 1–8. IEEE.
- Papachristos, C., Mascarich, F., Khattak, S., Dang, T., and Alexis, K. (2019b). Localization uncertainty-aware autonomous exploration and mapping with aerial robots using receding horizon path-planning. *Autonomous Robots*, 43(8):2131–2161.
- Petracek, P., Kratky, V., Petrlík, M., Baca, T., Kratochvil, R., and Saska, M. (2021). Large-Scale Exploration of Cave Environments by Unmanned Aerial Vehicles. *IEEE Robotics and Automation Letters*, 6(4):7596–7603.
- Petrlík, M., Baca, T., Hert, D., Vrba, M., Krajník, T., and Saska, M. (2020). A Robust UAV System for Operations in a Constrained Environment. *IEEE Robotics and Automation Letters*, 5(2):2169–2176.
- Petrlík, M., Krajník, T., and Saska, M. (2021). Lidar-based stabilization, navigation and localization for uavs operating in dark indoor environments. In *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 243–251. IEEE.

- Pritzl, V., Stepan, P., and Saska, M. (2021). Autonomous flying into buildings in a firefighting scenario. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 239–245. IEEE.
- Pritzl, V., Vrba, M., Stepan, P., and Saska, M. (2022a). Cooperative navigation and guidance of a micro-scale aerial vehicle by an accompanying uav using 3d lidar relative localization. In *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 526–535. IEEE.
- Pritzl, V., Vrba, M., Stepan, P., and Saska, M. (2022b). Cooperative navigation and guidance of a micro-scale aerial vehicle by an accompanying uav using 3d lidar relative localization. In *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 526–535. IEEE.
- Queralta, J. P., Taipalmaa, J., Pullinen, B. C., Sarker, V. K., Gia, T. N., Tenhunen, H., Gabbouj, M., Raitoharju, J., and Westerlund, T. (2020). Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision. *Ieee Access*, 8:191617–191643.
- Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
- Reinke, A., Palieri, M., Morrell, B., Chang, Y., Ebadi, K., Carlone, L., and Agha-Mohammadi, A.-A. (2022). Locus 2.0: Robust and computationally efficient lidar odometry for real-time 3d mapping. *IEEE Robotics and Automation Letters*, 7(4):9043–9050.
- Reynolds, D. A. (2009). Gaussian mixture models. *Encyclopedia of biometrics*, 741(659-663).
- Richter, C., Bry, A., and Roy, N. (2016). Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments. In *Robotics Research*, pages 649–666. Springer.
- Rogers, J. G., Gregory, J. M., Fink, J., and Stump, E. (2020a). Test your slam! the sub-tunnel dataset and metric for mapping. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 955–961.
- Rogers, J. G., Schang, A., Nieto-Granda, C., Ware, J., Carter, J., Fink, J., and Stump, E. (2020b). The darpa sub urban circuit mapping dataset and evaluation metric. In *International Symposium on Experimental Robotics*, pages 391–401. Springer.
- Roucek, T., Pecka, M., Cizek, P., Petricek, T., Bayer, J., Salansky, V., Hert, D., Petrlik, M., Baca, T., Spurny, V., et al. (2019). Darpa subterranean challenge: Multi-robotic exploration of underground environments. In *International Conference on Modelling and Simulation for Autonomous Systems*, pages 274–290. Springer.
- Roucek, T., Pecka, M., Cizek, P., Petricek, T., Bayer, J., Salansky, V., Hert, D., Petrlik, M., Baca, T., Spurny, V., Pomerleau, F., Kubelka, V., Faigl, J., Zimmermann, K., et al. (2020). Darpa subterranean challenge: Multi-robotic exploration of underground environments. In Mazal, J., Fagiolini, A., and Vasik, P., editors, *Modelling and Simulation for Autonomous Systems*, pages 274–290, Cham. Springer International Publishing.
- Saboia, M., Clark, L., Thangavelu, V., Edlund, J. A., Otsu, K., Correa, G. J., Varadharajan, V. S., Santamaria-Navarro, A., Touma, T., Bouman, A., et al. (2022). Achord: Communication-aware multi-robot coordination with intermittent connectivity. *arXiv preprint arXiv:2206.02245*.
- Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520.
- Santamaria-Navarro, A., Thakker, R., Fan, D. D., Morrell, B., and Agha-mohammadi, A.-a. (2019). Towards resilient autonomous navigation of drones. In *The International Symposium of Robotics Research*, pages 922–937. Springer.
- Scherer, S., Agrawal, V., Best, G., Cao, C., Cujic, K., Darnley, R., DeBortoli, R., Dexheimer, E., Drozd, B., Garg, R., et al. (2022). Resilient and modular subterranean exploration with a team of roving and flying robots. *Field Robotics*, 2:678–734.
- Shakhatreh, H., Sawalmeh, A. H., Al-Fuqaha, A., Dou, Z., Almaita, E., Khalil, I., Othman, N. S., Khreishah, A., and Guizani, M. (2019). Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges. *Ieee Access*, 7:48572–48634.
- Shan, T., Englot, B., Meyers, D., Wang, W., Ratti, C., and Daniela, R. (2020). Lio-sam: Tightly coupled lidar inertial odometry via smoothing and mapping. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5135–5142. IEEE.
- Shin, Y.-S., Park, Y. S., and Kim, A. (2020). Dvl-slam: sparse depth enhanced direct visual-lidar slam. *Autonomous Robots*, 44(2):115–130.
- Silano, G., Baca, T., Penicka, R., Liuzza, D., and Saska, M. (2021). Power line inspection tasks with multi-aerial robot systems via signal temporal logic specifications. *IEEE Robotics and Automation Letters*, 6(2):4169–4176.
- Smith, L. N. (2015). Cyclical learning rates for training neural networks.

- Spurny, V., Pritzl, V., Walter, V., Petrlik, M., Baca, T., Stepan, P., Zaitlik, D., and Saska, M. (2021). Autonomous firefighting inside buildings by an unmanned aerial vehicle. *IEEE Access*, 9:15872–15890.
- Tardioli, D., Riazuelo, L., Sicignano, D., Rizzo, C., Lera, F., Villarroel, J. L., and Montano, L. (2019). Ground robotics in tunnels: Keys and lessons learned after 10 years of research and experiments. *Journal of Field Robotics*, 36(6):1074–1101.
- Tomic, T., Schmid, K., Lutz, P., Domel, A., Kassecker, M., Mair, E., Grixia, I. L., Ruess, F., Suppa, M., and Burschka, D. (2012). Toward a fully autonomous uav: Research platform for indoor and outdoor urban search and rescue. *IEEE robotics & automation magazine*, 19(3):46–56.
- Tordesillas, J., Lopez, B. T., Everett, M., and How, J. P. (2022). Faster: Fast and safe trajectory planner for navigation in unknown environments. *IEEE Transactions on Robotics*, 38(2):922–938.
- Tranzatto, M., Dharmadhikari, M., Bernreiter, L., Camurri, M., Khattak, S., Mascarich, F., Pfreundschuh, P., Wisth, D., Zimmermann, S., Kulkarni, M., et al. (2022a). Team cerberus wins the darpa subterranean challenge: Technical overview and lessons learned. *arXiv preprint arXiv:2207.04914*.
- Tranzatto, M., Mascarich, F., Bernreiter, L., Godinho, C., Camurri, M., Khattak, S., Dang, T., Reijgwart, V., Loeje, J., Wisth, D., Zimmermann, S., Nguyen, H., Fehr, M., Solanka, L., Buchanan, R., Bjelonic, M., Khedekar, N., Valceschini, M., Jenelten, F., and Alexis, K. (2022b). Cerberus: Autonomous legged and aerial robotic exploration in the tunnel and urban circuits of the darpa subterranean challenge. *Field Robotics*, 2:274–324.
- Vrba, M., Hert, D., and Saska, M. (2019). Onboard marker-less detection and localization of noncooperating drones for their safe interception by an autonomous aerial system. *IEEE Robotics and Automation Letters*, 4(4):3402–3409.
- Walter, V., Novak, T., and Saska, M. (2018). Self-localization of unmanned aerial vehicles based on optical flow in onboard camera images. In *Lecture Notes in Computer Science, vol 10756.*, Cham. Springer International Publishing.
- Walter, V., Spurny, V., Petrlik, M., Baca, T., Zaitlik, D., Demkiv, L., , and Saska, M. (2022). Extinguishing real fires by fully autonomous multirotor uavs in the mbzirc 2020 competition. *Field Robotics*, 2:406–436.
- Williams, J., Jiang, S., O’Brien, M., Wagner, G., Hernandez, E., Cox, M., Pitt, A., Arkin, R., and Hudson, N. (2020). Online 3d frontier-based ugv and uav exploration using direct point cloud visibility. In *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pages 263–270. IEEE.
- Xu, W., Cai, Y., He, D., Lin, J., and Zhang, F. (2022). Fast-lio2: Fast direct lidar-inertial odometry. *IEEE Transactions on Robotics*.
- Zhang, J., Kaess, M., and Singh, S. (2016). On degeneracy of optimization-based state estimation problems. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 809–816. IEEE.
- Zhang, J. and Singh, S. (2014). Loam: Lidar odometry and mapping in real-time. In *Robotics: Science and Systems*, volume 2.
- Zhang, J. and Singh, S. (2015). Visual-lidar odometry and mapping: Low-drift, robust, and fast. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2174–2181. IEEE.
- Zhao, S., Zhang, H., Wang, P., Nogueira, L., and Scherer, S. (2021). Super odometry: Imu-centric lidar-visual-inertial estimator for challenging environments. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8729–8736.
- Zhou, B., Pan, J., Gao, F., and Shen, S. (2021a). Raptor: Robust and perception-aware trajectory replanning for quadrotor fast flight. *IEEE Transactions on Robotics*, 37(6):1992–2009.
- Zhou, B., Zhang, Y., Chen, X., and Shen, S. (2021b). Fuel: Fast uav exploration using incremental frontier structure and hierarchical planning. *IEEE Robotics and Automation Letters*, 6(2):779–786.

How to cite this article: Petrlík, M., Petráček, P., Krátký, V., Musil, T., Stasinchuk, Y., Vrba, M., Báča, T., Heřt, D., Pecka, M., Svoboda, T., & Saska, M. (2023). UAVs beneath the surface: Cooperative autonomy for subterranean search and rescue in DARPA SubT. *Field Robotics*, 3, 1–68.

Publisher’s Note: Field Robotics does not accept any legal responsibility for errors, omissions or claims and does not provide any warranty, express or implied, with respect to information published in this article.

4 Distributed Perception-Aware Multi-Robot Coordination

- [3c] **P. Petracek**, V. Kratky, M. Petrlik, T. Baca, R. Kratochvil, and M. Saska, “Large-Scale Exploration of Cave Environments by Unmanned Aerial Vehicles,” *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7596–7603, 2021

Another interdisciplinary overreach was made into the field of speleology in a core publication [3c] focusing on multi-robot exploration. This work was developed in close cooperation with the Institute of Geodesy at the Brno University of Technology. It was published in the *IEEE Robotics and Automation Letters* and was presented virtually at IEEE CASE 2021. Building on our previous works [8a, 12a], the work [3c] proposed a highly optimized self-contained UAV framework assisting human explorers, first responders, and speleologists in the mapping of natural cave systems. The contributions include full-stack LI-based UAV autonomy capable of autonomous operation in an arbitrary exploratory task within an unknown and unstructured subterranean environment for the purposes of accurate mapping. The experimental analyses achieved deep-exploratory flights up to a distance of 470 m in a single fully autonomous deployment in the Bull Rock Cave system (located in the central Moravian Karst, Czech Republic), requiring just 2 flights to map the entire cave system. Such long exploratory flights are rare, even in modern systems capable of autonomous real-world operation.

Apart from aerial autonomy, one of the primary contributions of the work [3c] is a novel strategy for distributed multi-robot homing. The strategy maximizes the flight time in tasks where homing of the deployed robots to an initial base location is not required. The proposed strategy is suitable for tasks where the possible information gain is superior to the cost of the robots, such as in S&R scenarios. To effectively use the limited flight time, the robots utilize intermittent local communication to plan the homing path, such that a group of robots can build up a communication tree with the base station (an initial takeoff position) as the root communication node. Each robot constructs a navigation homing tree using nodes created from its past poses, where a single tree node models a local sphere-modeled source of omnidirectional communication. The edges of the online-built tree are valued by the required flight time between two connected nodes, wherein the edge value is used to estimate the required homing time to the proximity of a communication node during exploration. The homing trees of different aerial explorers are integrated in a distributed manner once two robots are within the range of local communication. The homing-tree integration minimizes the homing time and maximizes the time capacity for the exploration task for robots that gain new information from the tree merging. The communication nodes are exclusively connected to build a retranslation chain. In contrast, the pose nodes are connected such that each path from a leaf to a communication node is the shortest with respect to the required flight time. Quantitative comparison of the results with a state-of-the-art baseline solution [135] has shown that a fifth deployed robot has a 21 % increase in the exploration time. The proposed method was deployed as the homing method of the UAV team during the DARPA SubT competition.

Large-Scale Exploration of Cave Environments by Unmanned Aerial Vehicles

Pavel Petráček^a, Vít Krátký^a, Matěj Petrlik^a, Tomáš Báča^a, Radim Kratochvíl^b, and Martin Saska^a

Abstract—This paper presents a self-contained system for the robust utilization of aerial robots in the autonomous exploration of cave environments to help human explorers, first responders, and speleologists. The proposed system is generally applicable to an arbitrary exploration task within an unknown and unstructured subterranean environment and interconnects crucial robotic subsystems to provide full autonomy of the robots. Such subsystems primarily include mapping, path and trajectory planning, localization, control, and decision making. Due to the diversity, complexity, and structural uncertainty of natural cave environments, the proposed system allows for the possible use of any arbitrary exploration strategy for a single robot, as well as for a cooperating team. A multi-robot cooperation strategy that maximizes the limited flight time of each aerial robot is proposed for exploration and search & rescue scenarios where the homing of all deployed robots back to an initial location is not required. The entire system is validated in a comprehensive experimental analysis comprising of hours of flight time in a real-world cave environment, as well as by hundreds of hours within a state-of-the-art virtual testbed that was developed for the DARPA Subterranean Challenge robotic competition. Among others, experimental results include multiple real-world exploration flights traveling over 470 m on a single battery in a demanding unknown cave environment.

Index Terms—Aerial Systems; Applications; Field Robots; Aerial Systems; Perception and Autonomy; Multi-Robot Systems; Mapping

MULTIMEDIA MATERIALS

The paper is supported by the multimedia materials available at mrs.felk.cvut.cz/papers/ral-2021-caves. The implementation is also publicly available at github.com/ctu-mrs.

I. INTRODUCTION

HUMAN exploration of complex cave systems has occurred for thousands of years. However, there are still entire cave systems and individual subterranean voids, shafts, and cavities that are yet uncovered. This is primarily due to the dangerous nature of subterranean exploration in environments like natural caves, although man-made cellars,

Manuscript received March 1, 2021; Revised May 6, 2021; Accepted July 8, 2021. This paper was recommended for publication by Editor Pauline Pounds upon evaluation of the Associate Editor and Reviewers' comments.

The work was supported by the Czech Science Foundation (GAČR) under research project no. 20-29531S, by CTU grant no. SGS20/174/OHK3/3T/13, by the Defense Advanced Research Projects Agency (DARPA), and by OP VVV funded project CZ.02.1.01/0.0/0.0/16 019/0000765 "Research Center for Informatics".

^a Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, 166 36 Prague 6, Czech Republic {pavel.petracek|vit.kratky|matej.petrlik|tomas.baca|martin.saska}@fel.cvut.cz,
^b Brno University of Technology, Faculty of Civil Engineering, Institute of Geodesy, Czech Republic r_kratochvil@fce.vutbr.cz.

Digital Object Identifier (DOI): see top of this page.



Fig. 1: Robotic exploration of the Bull Rock Cave (central Moravian Karst, Czech Republic) by a fully autonomous aerial vehicle.

drainages, and mines pose similar risks. These environments contain sediments such as debris, rocks, sand, clay, ice, decomposed organic matter, human waste, and even various forms of speleothems in limestone caves. Considering the absolute darkness, lack of GNSS signals, flowing and dripping water, humid air, and the possible presence of poisonous gases, wind gusts, hanging ropes, and wildlife, there is excessive risk to the lives of human explorers in the exploration of new environments, as well as in search & rescue missions. Given the current state-of-the-art technology in robotics, many dangerous areas of subterranean systems are safely reachable using mobile robots, with the greatest focus being on vertical exploration using aerial vehicles. In contrast to human exploration, the use of such technology presents several advantages in the form of accessibility, safety, speed, instantaneous environment visualization, and precise quantification. On the other hand, challenges to the operation of mobile robots in such an environment lies in the uncertainty, lack of light, high humidity, and diversity of space in the form of narrow and/or low passages, canyons, large domes, high chimneys, and deep abysses.

The challenges to deployment of aerial vehicles in subterranean environments with respect to robot control, communication, sensor fusion, and positioning are described thoroughly in [1]. These specific challenges continue to be relevant even after substantial progress in the field of mobile robotics. However, in contrast to [1], our motivation includes minimizing the need for communications required for operator control and instead focuses on the full autonomy of robots and autonomous cooperation among members of a robotic team. The restriction of communication in subterranean environments introduces challenges to the maximization of system robustness and the use of efficient decision making in the form of adaptable exploration strategies in harsh unknown environments.

A. Related work

In a non-robotic context, wild caves are explored by modernly termed *cavers*. However, the human surveying and mapping of caves is known to have existed for thousands of years for purposes ranging from dwelling to speleology. The significance of cave exploration and cave mapping to scientific research is a thoroughly studied inquiry in literature, e.g., in [2].

In the work presented here, we focus mainly on the robotic point of view within the scope of the application domain. One of the first cave-mapping approaches using robotic solutions was proposed in [3], where the authors employed hand-held laser scanners, which are limited in speed, accuracy, and safety. In the context of mobile robotics, topics like the automatic control of an unstable dynamic system such as an aerial multi-rotor vehicle [4], the fusion of inertial, visual, and laser information for localization and mapping [5], and path planning in dynamic environments [6] have been addressed in order to achieve faster and safer methodology than mapping done with hand-held devices, as proposed in [3].

Within the scope of subterranean environments, the DARPA Subterranean Challenge competition has pushed the state of the art of autonomous exploration in human-made mines [7]–[10]. Although these systems have provided interesting solutions with great potential, the authors of [7]–[10] rely on the predictable structure of underground mines, such as using the protraction of human-made tunnels to mark the furthest depth data as frontiers or predefining turns at junctions in [9]. Since the complexity and diversity of natural caves is extensive, more robust solutions with a minimum number of environmental assumptions are required. This was tackled in [11] where the authors introduced a possible way for applying autonomous drones as a technology to assist speleologists and archaeologists. Although an interesting read, the proposed methods only constitute a preliminary discussion that presents neither novel technology nor applied results. A similar discussion focusing on the state of robotic problems within the application of subterranean exploration with UAVs is presented in [12]. In contrast to [11], the authors of [12] present a set of preliminary experiments in laboratory conditions and two dimensional space. Unfortunately, the assumption of a planar world is highly restrictive within the scope of real-world deployment due to the complex character of natural subterranean environments.

The precise localization of mobile robots is crucial to autonomous navigation in such complex environments. Among existing state-of-the-art literature, the LOCUS algorithm [13] achieves the lowest localization error at the cost of high computational demands. Unlike with ground robots, this method might be unsuitable for aerial robots as the computational resources on lightweight UAVs are scarce due to their limited payload. In [14], the authors demonstrated that localization performance can be further improved by dropping range beacons. This is a viable strategy for heterogeneous robotic teams, but unfeasible for teams of only lightweight UAVs.

The use of robotic teams for cooperative exploration has been addressed mostly in planar worlds with recurrent connectivity constraints [15] or with the requirement of a centralized element [16]. A similarly defined task to our problem of team homing — respecting intermittent communication, need for

decentralization, and limited operation time of aerial robots — is proposed in [17], where the robots gather and share data during the mission and return all the way back to the base before their operation times out. In contrast to [17], we propose homing coordination that lands each aerial robot at a position expanding a communication relay graph, thereby increasing the time for mere exploration in tasks where return to the starting position is not required. Related to the scope of search & rescue, the authors in [18] propose to re-position robots in a relay-chain formation to enable data transmission over longer distances once an object of interest is found. Our solution reports the position of the objects once the explorer robot connects to the relay graph during homing. The recently developed fast exploration technique in [19] maximizes explored volume over battery-limited flight time. The method is based on data only from an RGBD camera with a limited field of view (FoV). In comparison to LiDAR-based methods, we have experimentally verified that RGBD cameras are sub-optimal sensors for the exploration of large-scale caves due to their limited range and FoV.

B. Contributions

First, we propose a fully autonomous system enabling multi-modal mapping, fast and efficient planning with sensoric field-of-view constraints for safe movement in 3D, robust localization, and adaptable decision making. Second, a multi-robot cooperation for the efficient homing of a team of autonomous explorer robots is proposed. Third, the system has been validated through hundreds of hours of testing in a state-of-the-art virtual testbed developed for the DARPA Subterranean Challenge robotic competition, as well as through hours of flight time in the real world. To the best of our knowledge, the presented large-scale experimental deployment of autonomous aerial robots in a natural cave environment goes beyond the current state of the art in autonomous robotics. Lastly, we present and share the experience obtained during this comprehensive experimental deployment that was carried out in close cooperation with speleologists.

II. EXPERIENCE GAINED

A. Speleology motivation

From the speleological point of view, aerial systems are crucial for pushing exploratory state-of-the-art methods to provide assistance in efficient scouting of difficult-to-access areas in vertical environments, as well as for the quick inspection of known areas using onboard sensors only. These systems minimize risks for humans by reducing the need to climb or to swim in cold water reservoirs, and also through the detection of poisonous gases or even radioactive waste. Furthermore, this enables the preservation and protection of natural environments against human influence, including ancient sediment forms, floor dripstone formations, paleontological and archaeological sites, and sources of potable water.

In contrast to well-established methods of subterranean documentation (i.e., theodolite and level/distance meter, compass, and clinometer), modern technology employs stationary and mobile laser scanners to produce a dense 3D model of the

environment. Due to the complexity of natural environments, the use of stationary scanners is time-consuming because of the necessity of eliminating occluded spaces. Although handheld mobile scanners are more time-efficient in this context, their use is limited to areas accessible to humans. This limitation opens the door for mobile robotics which is able to tackle this challenge and to provide optimized 3D mapping. State-of-the-art mapping in such environments reaches decimeter level precision, which is less precise than stationary scanners, yet sufficient for the majority of speleological needs. Moreover, the common issue of mapping drift accumulation in long-corridor spaces can be minimized using reference measurements by precise stationary scanners or man-measured control points to obtain accurate results.

B. System requirements

The primary prerequisite of a team of aerial explorers that can be deployed in caves involves the ability to adapt to diverse, unknown environments lacking sources of light and access to GNSS. This general description requires the abilities to

- be deployed in constrained cavities, as well as in open caverns of natural caves,
- map and visualize the environment in a fast, quantified manner in the form of dense point clouds and image streams,
- seamlessly infuse an arbitrary exploration strategy for more efficient mission operation within the scope of individual environments (policy selection is discussed in Sec. IV),
- return to the mission operator and promptly visualize the environment for human supervision, and
- maximize operation capabilities in terms of coverage when a team of robots is employed.

C. Depth estimation in high humidity

The performance of the PMD pico flexx time-of-flight (ToF) camera and the Intel Realsense D435 stereo camera have been analyzed as complementary sensors to the primary LiDAR for the purpose of improving the sensory FoV coverage. Although ToF cameras generally outperform stereo cameras in terms of distance measurement precision and density of measurement points [20], the high humidity typically present in natural caves causes dispersion of light emitted from ToF cameras by small water droplets. This effect significantly degrades the acquired measurements. As was verified empirically, ToF cameras can produce false-negative measurements of obstacles situated behind clouds of water droplets. The use of stereo cameras (e.g., Realsense) is recommended for its robustness to environmental conditions within natural caves. Nevertheless for large cave systems, such a sensor needs to be combined with 3D LiDARs in order to comply with the requirements of speleologists and first responders.

III. SYSTEM ARCHITECTURE

The system of the proposed autonomous explorer robot is divided into multiple groups of individual interconnected modules to be described in this section. All components and their relations are visualized in Fig. 2.

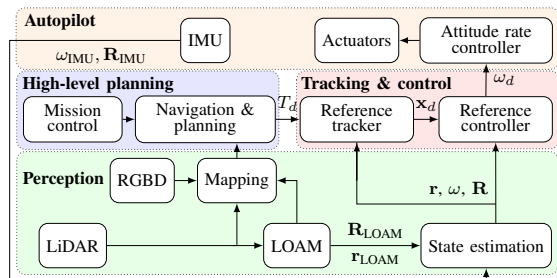


Fig. 2: Individual interconnected modules form the system architecture of the autonomous explorer robot. The *High-level planning* modules focus on achieving the mission objectives by generating references for the *Tracking & control* modules based on the map built by the *Perception* modules. This also provides a state estimate for closing the control feedback loop. All modules except the *Autopilot* group are handled by the main onboard computer.

A. Perception

The perception of the proposed system is based on a multi-channel *LiDAR* sensor that is used for both building the spatial representation of the surrounding environment in the *Mapping* module, as well as for the motion estimation in the *LOAM* module. Obtaining the full-state estimate is realized within the *State estimation* module, where multiple sources of incomplete state measurements are fused together using a bank-of-filters estimator.

The vertical navigation capabilities of the system can be greatly improved by equipping the robot with vertically-facing *RGBD* cameras that are able to fill in the blind spots in the limited vertical FoV of the *LiDAR*. Apart from navigation, these optional sensors may be used for detecting objects of interest in caves in search & rescue scenarios or for visual documentation of newly explored cave systems.

1) *LiDAR*: Even though our system is not tied to a specific LiDAR model, there are certain important parameters that can affect the performance and capabilities of the platform.

To reliably stabilize the UAV, the time delay of the estimated state must stay below the threshold of a certain critical value depending on the type of controller and gains. When this threshold is exceeded, the UAV begins oscillating and eventually automatically lands when the control error is too large to continue the mission safely. We have found experimentally that for most combinations of localization methods and controllers, the critical value ranges from 100 ms to 200 ms. Thus, 10 Hz is the lowest rotation frequency that can be used without employing methods of delay compensation.

The typical values of a vertical field of view (VFoV) of 3D LiDARs are in the 30° to 90° range. The higher VFoV values improve vertical mobility in constrained spaces, however with a low VFoV, it is impossible to safely navigate narrow vertical shafts as it is not known whether the space above the UAV is free and safe to fly through, or whether it contains an obstacle.

2) *RGBD*: The regions above and below the UAV that are not covered by the LiDAR can be captured using a depth camera or by spinning the LiDAR sensor around a vector that is orthogonal to the axis of scanning, as seen in [21]. However, such a solution adds additional weight to the sensor, which decreases the available flight time. A blind spot also still remains as part of the laser rays is blocked by the frame

of the UAV. Alternatively, lightweight depth cameras can be mounted on opposite sides of the body frame in order to cover most blind spots of the LiDAR. Additional sensing modality is gained by combining an RGB and depth camera in a single sensor (RGBD) with a slight weight increase.

3) *Localization*: For localization of the UAV, we have adapted the LOAM algorithm [22]. This state-of-the-art method is very precise (0.55% translation error [23]) while attaining real-time performance. In our adapted version of the open-source implementation, the algorithm is optimized on CPU and employs parallel computing, which enables us to deploy and use the localization in the real-time position control feedback loop onboard fast-moving aerial vehicles.

4) *Mapping*: The LOAM-based algorithm builds a sparse internal representation of the environment consisting of edge and planar features. However, this sparse map is unsuitable for navigation purposes. Additionally, the LOAM map does not consider the probabilistic nature of the sensor, nor does it distinguish free and unknown space. Both of these factors are necessary in exploration techniques for reliable navigation and consistent frontier selection.

In the proposed system, the environment is represented by a dense probabilistic volumetric map, which consists of cubic cells with one of 3 states: free, occupied, or unknown. The map is kept in the octree structure to facilitate the Bayesian integration of new measurements and efficient access to individual cells of the probabilistic map. The high-level systems, such as grid-based path planning or inter-robot map registration, also benefit from quick access to the dense environment representation. This approach is capable of multi-modal fusion by integrating the data from all available onboard sensors and outputting point-cloud measurements. If high-level path planning is constrained by the field of view of onboard sensors (tackled in [24] and also in Sec. III-C), the multi-modality of mapping enables arbitrary movement in 3D.

5) *Sensor processing*: The targeted subterranean environments may have high humidity or may contain large clouds of whirling dust. The water and dust particles can then produce erroneous measurements for the LiDAR-based sensors. Assuming a partial reflection from water or dust particles and a large energy dissipation of distant reflections, these erroneous measurements can be filtered with respect to the measured intensity of returning light rays. As has been empirically verified, a simple threshold-based filtration over the intensity channel within the local neighborhood of the sensor is sufficient for filtering out false-positive measurements. The idea of the local filtration is to filter out particles gusting through the surrounding air due to the aerodynamic influence of the propellers. Although the cutoff threshold of the intensity magnitude is environment-specific, filtering out measurements below the 10th percentile of the intensity distribution per each laser scan proved to be a reliable solution, even in the dustiest real-world environments. Such processing is unavailable for camera-based systems that may require thorough, computationally-expensive solutions to overcome these challenges.

6) *State estimation*: The reference controller (see Sec. III-B2) requires a position estimate of the UAV body frame in the world frame $\mathbf{r} = (x, y, z)$, the velocity of the

body frame $\dot{\mathbf{r}}$, rotation \mathbf{R} from the UAV body frame to the world frame, and angular velocity $\boldsymbol{\omega}$ in the body frame in order to close the feedback loop. The LOAM localization method provides 6-DoF pose estimate, i.e., \mathbf{r}_{LOAM} , \mathbf{R}_{LOAM} , which are fused in the *State estimation* block with interoceptive measurements from the *IMU* of the *Autopilot* to obtain the rest of the state variables.

The details about the estimation process are described in [25]. Nevertheless, it is worth highlighting the importance of the fusion of orientation \mathbf{R}_{IMU} and \mathbf{R}_{LOAM} in cave environments. While \mathbf{R}_{IMU} is very precise and without delay, the heading of the UAV (i.e., the measured direction of the body-fixed, forward-facing axis) is unreliable due to the presence of ferromagnetic ores in the cave rocks that cause deviations in the magnetometer measurements. By correcting these errors with the heading from \mathbf{R}_{LOAM} in the estimation process, the resulting orientation \mathbf{R} is robust to changes in the erratic magnetic field in subterranean environments.

B. Tracking & control

The safe navigation of constrained environments with low obstacle clearance imposes the requirements of precise trajectory tracking with minimal control error, as any deviation from the desired state could potentially result in a collision. The *Reference controller* is responsible for minimizing the control error around the desired control reference that is provided by the *Reference tracker*. The controller outputs an attitude rate reference for the low-level *Attitude rate controller* in the *Autopilot*.

1) *Reference tracker*: The *Reference tracker* is essential in providing the *Reference controller* with smooth and feasible references to ensure a safe flight. The tracker based on the model predictive control (MPC) simulates an ideal virtual model of the UAV with constrained translational states up to jerk, together with heading and heading rate. The input can be either a single pair of desired 3D position \mathbf{p}_d and heading η_d , or a trajectory T_d in the form of a sequence of such pairs with a specified sampling rate. The full state of the virtual model is then sampled at 100 Hz, and \mathbf{r}_d , $\dot{\mathbf{r}}_d$, $\ddot{\mathbf{r}}_d$, $\dot{\eta}_d$, η_d , $\dot{\eta}_d$ are passed to the *Reference controller* as reference \mathbf{x}_d .

2) *Reference controller*: The agile *SE(3)* geometric state feedback controller [26] minimizes the position and velocity errors. To compensate imperfect calibration and external forces acting upon the UAV, the controller is extended with the body and world disturbance terms described in [25]. The output attitude rate reference $\boldsymbol{\omega}_d$ is tracked by the *Autopilot*.

C. Path planning

The planning approach used to safely navigate through a priori unknown environments must fulfill requirements of real-time responsiveness and efficient global planning in order to fully exploit the limited flight time of UAVs. For this purpose, fast iterative post-processing is applied to the output of an optimal grid-based planner in order to increase the UAV-obstacle distance above a minimum threshold [27]. The grid-based planner and the iterative post-processing do not apply an optimistic assumption that the unknown space is collision-free. Although this visibility-constrained precondition requires

high sensory coverage around the robot to allow for arbitrary movement in 3D, it consequently prevents collisions of the trajectory being followed, even if replanning would fail. This methodology improves safety and robustness of the overall flight, allows for deployment in completely unknown environments without any apriori information, and permits seamless navigation in open spaces, as well as safe movement through narrow passages.

Common grid-based planning methods require pre-processing of an employed map representation, such as determining and applying the 3D distance transform for obstacle growing. This may introduce significant computational overhead by bottle-necking system performance, as the map must then be processed in every planning step. Such a computationally expensive task contradicts the requirements for responsiveness within evolving dynamic environments. To minimize the overall time required for a single planning iteration, a local KD-tree representation of the environment is used to decide the feasibility of particular cells within a voxel grid. This approach shifts the largest load from the pre-processing phase to the planning phase, which is beneficial especially to shorter plans that require searching only a small part of the environment. The low computational demands of the applied planning approach enable frequent replanning the global plan, which is also crucial for the efficient use of newly-discovered collision-free space.

To effectively exploit the limited flight time of aerial explorers, all mid-flight stops are eliminated by computing in parallel the next exploration goal during path following. The path to the next goal is efficiently appended to the rest of the current reference trajectory T_d using the prediction horizon of the MPC (see Sec. III-B). The need for precise locomotion control in complex natural caves makes uniform path-sampling unfeasible with respect to the dynamic constraints of a UAV and fast, collision-free trajectory tracking. Therefore, the reference trajectory T_d provided by the *Navigation & planning* module to the *Reference tracker* is computed based on the following process.

Given the dynamical constraints of the robot, the generated path is uniformly sampled with a sampling distance adapted to the maximum velocity magnitude v_{max} of the UAV. Based on this initial trajectory T_i , the required acceleration magnitudes a_n between consequent transition points are computed by velocity differentiation as

$$a_n(k) = \frac{\|\mathbf{v}_i(k+1) - \mathbf{v}_i(k)\|_2}{t_s}, \quad (1)$$

where $\mathbf{v}_i(k)$ is the required velocity vector for transition from a transition point $t_i(k)$ to $t_i(k+1)$ on the initial trajectory T_i and t_s is a constant sampling period. The new velocity for a k -th segment is then given by

$$v_k = \begin{cases} \max\left(v_{max} \frac{a_{max}}{a_n}, v_{min}\right) & \text{if } a_n(k) > a_{max}, \\ v_{max} & \text{if } a_n(k) \leq a_{max}, \end{cases} \quad (2)$$

where the minimum velocity v_{min} serves as a parameter balancing the precision and the time needed for trajectory tracking. By this step, the velocities for particular segments are set so that the maximum velocity is applied in straight segments, while lower velocities are applied in curved segments of any

given path.

To further improve trajectory sampling and to achieve smoother changes in velocities, the sampling distance on particular segments is computed so that the motion along each segment has the constant acceleration

$$\bar{a}_k = \frac{|v_{k+1} - v_k|}{t_{acc,k}}, \quad (3)$$

where $t_{acc,k}$ is the time available for acceleration on the k -th segment. The time $t_{acc,k}$ is obtained from the length l_k of the segment k and the required change of the velocity. The number of transition points N_k on the k -th segment of the initial trajectory T_i is given as

$$N_k = \begin{cases} \left\lceil \frac{l_k}{v_k t_s} \right\rceil & \text{if } \bar{a}_k = 0, \\ \left\lceil \frac{t_{acc,k}}{t_s} \right\rceil & \text{if } \bar{a}_k > 0, \end{cases} \quad (4)$$

where the desired constant acceleration is adapted to meet the velocity v_{k+1} at the end of each segment as

$$a_k = \frac{\bar{a}_k}{N_k t_s}. \quad (5)$$

The sequence of sampling distances for the k -th segment of T_i is then given by

$$d_{k,i} = v_k t_s + i a_k t_s^2, \quad i \in \{1, \dots, N_k\}. \quad (6)$$

The trajectory sampled with sampling distances defined by (6) is passed to the *Reference tracker* [25] as a reference trajectory T_d in order to generate a feasible reference \mathbf{x}_d for the *Reference controller*. Despite its simplicity, the described sampling method achieves better results within the scope of the proposed application than the optimization-based trajectory generation methods proposed in [28], [29]. In contrast to the proposed method, the problem in [28], [29] is defined in such a way that the exact positions of all the path waypoints must be visited, generating significantly slower trajectories.

IV. EXPLORATION POLICY

Cave environments are naturally diverse and require various different mission strategies suitable for specific environments. Deriving the optimal policy is thereby dependent on various factors, such as the expected mission output, mission-specific constraints, the complexity and the specifics of the environment, and the number of available robots. For this reason, our system is designed so that any arbitrary policy can be utilized within the scope of an autonomous mission.

Nevertheless, two exploratory mission types are of the most use in practice: deep cave exploration and full-coverage exploration. These missions are used for scouting previously uncovered areas in order to obtain a general overview of the environment, monitor environmental changes such as gas leaks, detect natural water reservoirs, discover new possible passages, or assess the structural state of cavern walls and other objects of interest. The former approach maximizes the explored volume of space in the entire environment, while the latter minimizes the blind spots missed by onboard cameras with a constrained FoV.

The capabilities of a robotic mission are furthered with the use of multiple cooperating robots. To show an example of such improvement using a team of agents as opposed to a

single agent, a homing strategy that maximizes the flight time of aerial robots during a multi-robotic mission is proposed in the following subsection. During the proposed coordination, continuous exploration is not assumed and distance-constrained ad-hoc communication is used. The robots are homogeneous and generate their behaviors in a decentralized manner based on their current state and the available information from other robots (only positions in a shared frame are required).

A. Multi-robot homing strategy

A cooperative operation maximizing the flight time of a multi-robot team is proposed for applications where homing all the deployed robots to an initial location is not required. This strategy is suitable for tasks where the possible gained information is superior to the cost of the robots, such as in search & rescue scenarios. This method assumes there is access to a low-bandwidth communication link among any two robots within an omnidirectional communication radius.

To maximize the flight time, the robots utilize local communication to plan the homing path such that a group of robots is able to build up a communication tree with the base station as the root communication node. This allows the robots to optimize their flight time by navigating back to a location in the proximity of another communication node (a landed robot, base station, or self-sustaining communication node deployed by other robots) when the battery capacity becomes drained. This entire homing strategy is showcased in an example scenario for two independent robots in Fig. 3.

In the proposed strategy, each robot constructs a navigation homing tree using nodes created from the set of past poses of the robot. This online-built tree has edges valued by the required flight time between two nodes and is used to estimate required homing time to the proximity of a communication node. The pose nodes are connected such that each path leaf-to-communication is the shortest (see Fig. 3a). A homing path is constructed recursively as a sequence of tree nodes from the current robot position (a leaf) to the nearest communication node, with the landing position being within communication range of the nearest communication node (see Fig. 3b). The tree is shared among the robots deployed in the same mission. The knowledge from the previous explorers is integrated to prolong their flight time (see Fig. 3c), thus causally maximizing the time capacity for the exploration task. When a communication node (e.g., a robot landing pose) is integrated into the homing tree, it is linked exclusively to another communication node to join the retranslation chain (see Fig. 3d). Consequently, the parents of neighboring pose nodes are updated so that each pose node has a parent with the minimal accumulated cost to any communication node (see Fig. 3b and Fig. 3d). The process of inserting pose nodes as well as communication nodes into the homing tree is described in Alg. 1.

V. EXPERIMENTAL ANALYSIS

The entire proposed system has been validated through hours of flight time in the real world, as well as in hundreds of hours in various virtual subterranean environments. The results of these experimental analyses are presented hereafter.

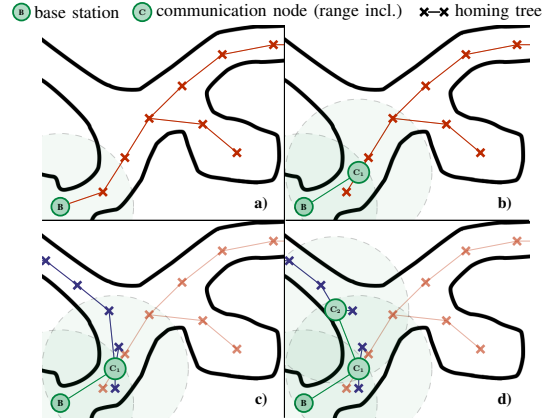


Fig. 3: An example scenario of the homing strategy for two robots (red and blue) that maximizes flight time by landing at feasible positions while building a communication chain to a base station.

Algorithm 1: Insertion of a node into the onboard-built homing tree. Function $\text{cost}(n_a, n_b)$ returns an estimate of flight time among nodes n_a and n_b , function $\text{accumulatedCost}(n_a)$ returns the required flight time from node n_a to the nearest communication node, and function $\text{freeRay}(n_a, n_b)$ returns true if a linear path between nodes n_a and n_b is collision-free in 3D.

```

1: procedure INSERTNODETOHOMINGTREE
2:   Input:
3:      $N$                                       $\triangleright$  Node to be inserted
4:      $C, \mathcal{P}$                                 $\triangleright$  Sets of communication and pose nodes
5:      $d_e$                                     $\triangleright$  Minimum edge length
6:   if  $N.type == \text{COMMUNICATION}$  then
7:      $N.parent \leftarrow \arg \min_{c \in C} \text{cost}(N, c)$ 
8:     for  $p \in \mathcal{P}$  do                        $\triangleright$  Update parents of neighboring pose nodes
9:       if  $\text{cost}(N, p) < \text{accumulatedCost}(p)$  then
10:         $p.parent \leftarrow N$ 
11:      $C \leftarrow C \cup N$ 
12:   else
13:      $\mathcal{V} \leftarrow C \cup \mathcal{P}$ 
14:     if  $\min_{v \in \mathcal{V}} (\|N - v\|_2) \geq d_e$  then
15:        $\bar{\mathcal{V}} = \{v \mid \text{freeRay}(N, v), \forall v \in \mathcal{V}\}$ 
16:       if  $\bar{\mathcal{V}} \neq \emptyset$  then
17:          $N.parent \leftarrow \arg \min_{v \in \bar{\mathcal{V}}} [\text{cost}(N, v) + \text{accumulatedCost}(v)]$ 
18:          $\mathcal{P} \leftarrow \mathcal{P} \cup N$ 

```

A. Real-world environment

To analyze the properties of the system, a fully autonomous aerial robot (see Fig. 4) was deployed for several hours of flight time in the Bull Rock Cave located in the central Moravian Karst of the Czech Republic (see Fig. 1 and the attached multimedia materials).

During multiple autonomous exploratory missions, a single explorer (see the hardware components of the robot in Fig. 4) was deployed to validate the proposed system in various exploratory scenarios. The flight trajectories from all missions are visualized in Fig. 5a and the mission statistics and performance metrics of the mapping module are summarized in Table I. A greedy frontier-navigation policy was employed such that the frontier closest to the lateral direction of flight (A, B), the highest frontier (C), and frontier with the largest ratio of unknown to free cells in a bounded area (D) was selected as the next goal. With respect to these experiments in a harsh subterranean environment, we have

- validated the performance of the system by flying in large cave domes, as well as in narrow corridors just 70 cm wider than the dimensions of the robot,

Trial	Flight time (s)	Trajectory length (m)	Explored volume (m ³)	Mapping accuracy	
				μ (m)	σ (m)
A	731	476	7463	0.57	0.59
B	935	473	11 403	0.53	0.56
C	359	71	551	0.23	0.26
D	749	602	3756	0.33	0.38
E	386	233	3055	0.39	0.39
F	633	256	2579	0.21	0.22
G	638	261	3650	0.27	0.33
H	297	142	1682	0.25	0.41
I	129	121	3326	0.19	0.22
J	425	233	4388	0.25	0.29

TABLE I: Quantitative evaluation on multiple autonomous exploratory missions within the Bull Rock Cave system. The flight trajectories and qualitative analysis of the mapping accuracy are shown in Fig. 5.

- validated the real-time performance and robustness of the system in multiple autonomous horizontally-deep flights longer than 470 m using just a single battery and reaching a maximal velocity up to 2 m s^{-1} ,
- validated the ability to autonomously explore natural domes in terms of vertical depth,
- verified the ability to perform a full mission and return to an initial location with the obtained information,
- quantified the accuracy of the onboard-built maps with respect to a ground truth map of the environment, and
- obtained feedback from speleologists in order to design the system following their requirements.

The dense onboard-built maps (20 cm resolution) from all the experiments were merged (manual global registration with local ICP refinement) during post-processing to obtain the map of the environment \mathcal{M} . The reference ground truth map \mathcal{M}_{gt} was built by registering over 100 largely overlapping scans taken by a Leica BLK360 terrestrial 3D scanner. The mapping accuracy over all the experiments reached mean $\mu = 0.37 \text{ m}$ and standard deviation $\sigma = 0.46 \text{ m}$ using the point-to-point Euclidean error metric between each point in \mathcal{M} and the corresponding closest point in \mathcal{M}_{gt} . The distribution of the mapping errors throughout all flights is visualized in Fig. 5b. As specified by the end-users, the decimeter-level mapping precision achieved over the course of these exceptionally fast and extensive flights is sufficient for the majority of speleological needs.

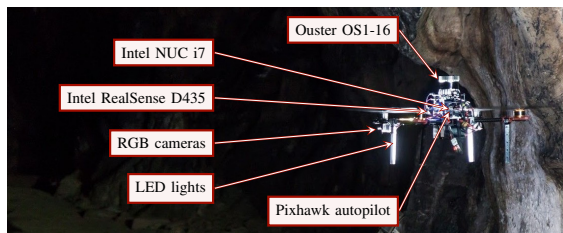
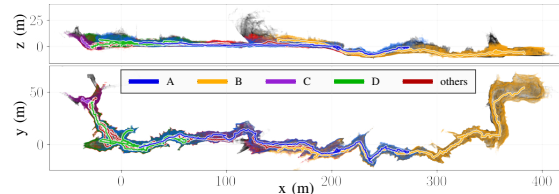


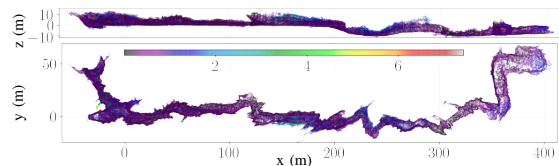
Fig. 4: General hardware components of an autonomous explorer robot. All data are processed and reasoned over with an onboard processing unit. The main source of data comes from the top-mounted LiDAR.

B. Virtual environment

To validate the proposed methodology for multi-robot coordination using a local low-bandwidth communication network, a team of aerial robots was deployed for hundreds of hours of flight in a virtual environment using a virtual testbed developed for the DARPA Subterranean Challenge competition.



(a) Overview of the cave environment with the trajectories of all exploration missions (see Table I) performed within Bull Rock Cave. The figure shows deep cave missions (A, B), vertical flight (C), and the thorough exploration of a bounded area (D).



(b) Visual analysis on the mapping accuracy – the distribution of mapping errors during all autonomous exploration tasks as summarized in Table I. The color bar legend represents the mapping error in meters using the point-to-point euclidean error metric.

Fig. 5: Full-coverage exploration of the Bull Rock Cave system (located in the central Moravian Karst, Czech Republic) with autonomous aerial explorers. Full resolution figure is available within the attached multimedia.

This state-of-the-art testbed consists of several large-scale cave environments containing dynamic obstacles and models of real-world interference, such as sensor discrepancies, communication schemes, and battery longevity.

In contrast to real-world experiments, the virtual environment is larger and allows for the seamless verification of multi-robotic cooperation. To demonstrate the performance of the proposed homing strategy, a selected example scenario of such an operation is presented in Fig. 6. This experiment highlights the positive influence of the homing strategy in a search & rescue scenario where the three explorers were able to exploit the increased flight time. With a 50 m communication range and 1.2 m s^{-1} average velocity for each robot, the homing cooperation increased the available flight time for exploration by 40 s and 80 s, respectively. Moreover, the experiment shows the influence of multi-sensor mapping, which allowed the black robot to single-handedly explore the upper floor of the virtual environment. The final exploratory trajectories of the cooperating robots during the presented mission reached lengths of 715 m, 1349 m, and 1405 m.

The influence of the homing strategy on the time available for mere exploration is also quantitatively analyzed in Table II. The results were averaged over six separate deployments, each with five cooperating robots. Identical mission parameters were set to all the robots for the baseline [17], as well as for the proposed method. The data show an increasing trend in the available mission time for belated explorers for which the effective exploration phase is consequently prolonged during their entire operation time.

VI. CONCLUSION

This letter presents a comprehensive study on the use of autonomous aerial explorers as an assisting technology for the exploration of natural cave environments. This study also

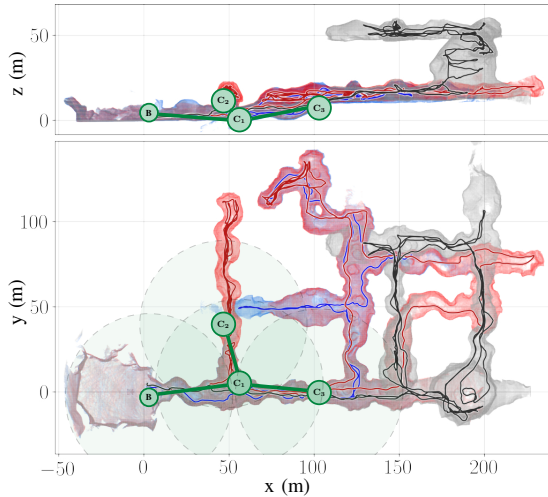


Fig. 6: Three autonomous explorers deployed in a virtual cave world within the DARPA simulation testbed. The robots finished their missions by building a communication tree with maximal edge length of $d_c = 50$ m.

Robot	1 st	2 nd	3 rd	4 th	5 th
Exploration time before homing (secs)	316	330	360	380	389
Exploration time increase (%)	-1.5	2.8	12.2	18.4	21.0

TABLE II: Influence of the homing strategy on the flight time available for mere exploration. Comparison with a baseline time of 321 s (averaged over 10 flights) where a robot returned to base before its operation timed out.

shares the experience acquired during the technology's development in close cooperation with a team of speleologists, cavers, and first responders.

The proposed self-sustaining system interconnects solutions for all crucial robotic tasks in order to enable full autonomy in complex unknown subterranean environments without access to GNSS. Among others, this includes laser-data processing which copes with high humidity and dustiness within subterranean environments and robust path-planning for unknown dynamic environments to allow for flights in constrained cavities, as well as in open caverns of natural caves. Moreover, a multi-robot cooperation is proposed for the efficient homing of a team of robots for applications where the possible information gain is superior to the costs of the robots, such as search & rescue scenarios in cave systems. The performance of the entire applicable system was validated in one of the most large-scale experimental analyses ever conducted, consisting of hours of flight time in Bull Rock Cave (Czech Republic, Moravian Karst) and in hundreds of hours in the state-of-the-art virtual testbed developed for the DARPA Subterranean Challenge. This presented analysis of the entire system proves that it is a robust solution capable of reliable planning with sensoric field-of-view constraints and accurate mapping. The accuracy of localization and mapping was evaluated with respect to a ground-truth map of the cave environment and reached mean precision below 40 cm in real-world conditions. This performance has satisfied the requirements of speleologists and first responders.

REFERENCES

- [1] R. La Scalea *et al.*, "Opportunities for autonomous UAV in harsh environments," in *IEEE ISWCS*, 2019, pp. 227–232.
- [2] P. Kambesis, "The importance of cave exploration to scientific research," *Journal of Cave and Karst Studies*, vol. 69, pp. 46–58, 04 2007.
- [3] R. Zlot *et al.*, "Three-Dimensional Mobile Mapping of Caves," *Journal of Cave and Karst Studies*, vol. 76, pp. 191–206, 2014.
- [4] T. P. Nascimento *et al.*, "Position and attitude control of multi-rotor aerial vehicles: A survey," *Annual Reviews in Control*, vol. 48, pp. 129–146, 2019.
- [5] C. Debeunne *et al.*, "A Review of Visual-LiDAR Fusion based Simultaneous Localization and Mapping," *Sensors*, vol. 20, no. 7, p. 2068, 2020.
- [6] Y. Zhao *et al.*, "Survey on computational-intelligence-based UAV path planning," *Knowledge-Based Systems*, vol. 158, pp. 54–64, 2018.
- [7] K. Ebadi *et al.*, "LAMP: Large-Scale Autonomous Mapping and Positioning for Exploration of Perceptually-Degraded Subterranean Environments," in *IEEE ICRA*, 2020, pp. 80–86.
- [8] T. Dang *et al.*, "Explore Locally, Plan Globally: A Path Planning Framework for Autonomous Robotic Exploration in Subterranean Environments," in *IEEE ICAR*, 2019, pp. 9–16.
- [9] I. D. Miller *et al.*, "Mine Tunnel Exploration Using Multiple Quadrupedal Robots," *IEEE RA-L*, vol. 5, no. 2, pp. 2840–2847, 2020.
- [10] M. Petrlik *et al.*, "A Robust UAV System for Operations in a Constrained Environment," *IEEE RA-L*, vol. 5, no. 2, pp. 2169–2176, 2020.
- [11] G. Zhang *et al.*, "SmartCaveDrone: 3D cave mapping using UAVs as robotic co-archaeologists," in *IEEE ICUAS*, 2017, pp. 1052–1057.
- [12] J. G. R. III *et al.*, "Distributed subterranean exploration and mapping with teams of UAVs," in *Ground/Air Multisensor Interoperability, Integration, and Networking for Persistent ISR VIII*, vol. 10190, 2017, pp. 285–301.
- [13] M. Palieri *et al.*, "LOCUS: A Multi-Sensor Lidar-Centric Solution for High-Precision Odometry and 3D Mapping in Real-Time," *IEEE RA-L*, vol. 6, no. 2, pp. 421–428, 2020.
- [14] N. Funabiki *et al.*, "Range-Aided Pose-Graph-Based SLAM: Applications of Deployable Ranging Beacons for Unknown Environment Exploration," *IEEE RA-L*, vol. 6, no. 1, pp. 48–55, 2020.
- [15] Y. Pei *et al.*, "Coordinated Multi-Robot Real-Time Exploration with Connectivity and Bandwidth Awareness," in *IEEE ICRA*, 2010, pp. 5460–5465.
- [16] W. Burgard *et al.*, "Coordinated Multi-Robot Exploration," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 376–386, 2005.
- [17] K. Cesare *et al.*, "Multi-UAV Exploration with Limited Communication and Battery," in *IEEE ICRA*, 2015, pp. 2230–2235.
- [18] J. Scherer *et al.*, "An Autonomous Multi-UAV System for Search and Rescue," in *DroNet*, 2015, p. 33–38.
- [19] B. Zhou *et al.*, "FUEL: Fast UAV Exploration Using Incremental Frontier Structure and Hierarchical Planning," *IEEE RA-L*, vol. 6, no. 2, pp. 779–786, 2021.
- [20] F. Pece *et al.*, "Three depth-camera technologies compared," in *First BEAMING Workshop, Barcelona*, vol. 2011. Citeseer, 2011, p. 9.
- [21] J. Williams *et al.*, "Online 3D Frontier-Based UGV and UAV Exploration Using Direct Point Cloud Visibility," in *IEEE MFI*, 2020, pp. 263–270.
- [22] J. Zhang *et al.*, "Low-drift and real-time lidar odometry and mapping," vol. 41, no. 2, 2017, pp. 401–416.
- [23] A. Geiger *et al.*, "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite," in *IEEE CVPR*, 05 2012, pp. 3354–3361.
- [24] M. Beul *et al.*, "Fast Autonomous Flight in Warehouses for Inventory Applications," *IEEE RA-L*, vol. 3, no. 4, pp. 3121–3128, 2018.
- [25] T. Baca *et al.*, "The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles," *J Intell Robot Syst*, vol. 102, no. 26, pp. 1–28, 2021.
- [26] T. Lee *et al.*, "Geometric tracking control of a quadrotor UAV on SE(3)," in *IEEE CDC*, 2010, pp. 5420–5425.
- [27] V. Krátký *et al.*, "An Autonomous Unmanned Aerial Vehicle System for Fast Exploration of Large Complex Indoor Environments," *JFR*, vol. 5, no. 2, pp. 1–24, 2021.
- [28] C. Richter *et al.*, "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments," in *Robotics Research*. Springer, 2016, pp. 649–666.
- [29] M. Burri *et al.*, "Real-Time Visual-Inertial Mapping, Re-localization and Planning Onboard MAVs in Unknown Environments," in *IEEE IROS*, Sept 2015, pp. 1872–1878.

- [5c] **P. Petracek**, V. Walter, T. Baca, and M. Saska, “Bio-Inspired Compact Swarms of Unmanned Aerial Vehicles without Communication and External Localization,” *Bioinspiration & Biomimetics*, vol. 16, no. 2, p. 026 009, 2021

Our endeavor in improving robustness and fault tolerance in multi-robot teams has culminated in the core publication [5c], which was published in *Bioinspiration & Biomimetics*. To tackle the challenges, the work extends theoretical and practical contributions in the field of perceptually-aware swarming. The contributions include a tightly coupled perception-action flocking method capable of local self-organization and global navigation of an arbitrary number of UAVs which do not communicate among themselves and sense only locally in an environment with or without obstacles. The work proposes a decentralized framework built on a biologically-inspired swarming model capable of dynamic aggregation, separation, alignment, navigation, and obstacle avoidance, while requiring only local relative information. In a distributed manner, each agent reacts exclusively to its current perceptual input by computing an action based on a combination of carefully designed rules — one for each of the desired global behaviors. The cohesion, separation, and alignment rules are inspired by one of the first models in the field [96], which was designed for dimensionless particles. The model was adapted for real-world use with nonlinear weighting functions and scale factors derived from the rates of real-time systems applied in the control feedback loop of a UAV. The obstacle avoidance utilizes ideas of virtual agents similar to [97] but classifies proximal obstacles to two geometric classes (circles and lines) detectable by a lightweight 2D LiDAR on board. The applicability of the work in the real world is achieved with the UVDAR [37, 38] relative localization system, which perceives relative positions of UAVs in a robot’s proximity.

The work [5c] is unique for its achievement of fundamental swarm properties as defined in [36] in not only in simulation analyses but also in real-world non-laboratory conditions. At the publication date, this research included the first experimental deployments and performance studies of decentralized compact swarms of UAVs capable of navigation through obstacle-filled environments without any communication or information sharing. Beyond these contributions, [5c] presents a theoretical discussion on the framework’s swarm scalability (i.e., on the upper limit of the number of agents) and defines general scalability maximization in visually-aware swarms. The theoretical contributions of [5c] also include an analysis of the relationship between the accuracy of relative position estimation and swarm stability. These results show that a zero-mean normal error in the estimated position induces robot-to-robot oscillations with an exponentially growing magnitude correlating with the deviation of the error distribution.

The developed ideas, designs, and methods of [5c] have served and continue to serve as a base to the state-of-the-art research of fully decentralized swarming within the MRS laboratory. To this day, this research topic has resulted in several publications written in the MRS laboratory and published in impacted journals and conference proceedings [11a, 14a–16, 117, 118, 136].

Bioinspiration & Biomimetics



PAPER

Bio-inspired compact swarms of unmanned aerial vehicles without communication and external localization

RECEIVED
31 May 2020

REVISED
31 August 2020

ACCEPTED FOR PUBLICATION
2 November 2020

PUBLISHED
18 December 2020

Pavel Petráček^{*}, Viktor Walter, Tomáš Báča and Martin Saska

Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, 166 36, Prague 6, Czech Republic
* Author to whom any correspondence should be addressed.

E-mail: pavel.petracek@fel.cvut.cz, viktor.walter@fel.cvut.cz, tomas.baca@fel.cvut.cz and martin.saska@fel.cvut.cz

Keywords: swarm robotics, relative localization, distributed control, unmanned aerial vehicle

Abstract

This article presents a unique framework for deploying decentralized and infrastructure-independent swarms of homogeneous aerial vehicles in the real world without explicit communication. This is a requirement in swarm research, which anticipates that global knowledge and communication will not scale well with the number of robots. The system architecture proposed in this article employs the ultraviolet direction and ranging technique to directly perceive the local neighborhood for direct mutual localization of swarm members. The technique allows for decentralization and high scalability of swarm systems, such as can be observed in fish schools, bird flocks, or cattle herds. The bio-inspired swarming model that has been developed is suited for real-world deployment of large particle groups in outdoor and indoor environments with obstacles. The collective behavior of the model emerges from a set of local rules based on direct observation of the neighborhood using onboard sensors only. The model is scalable, requires only local perception of agents and the environment, and requires no communication among the agents. Apart from simulated scenarios, the performance and usability of the entire framework is analyzed in several real-world experiments with a fully-decentralized swarm of unmanned aerial vehicles (UAVs) deployed in outdoor conditions. To the best of our knowledge, these experiments are the first deployment of decentralized bio-inspired compact swarms of UAVs without the use of a communication network or shared absolute localization. The entire system is available as open-source at <https://github.com/ctu-mrs>.

1. Introduction

Use of a team instead of a single robot may yield several general advantages in tasks that either benefit from the multi-robot configuration or are altogether unsolvable by a single robot. The main advantages of robot teams are reduced task execution time, improved robustness, redundancy, fault tolerance, and convenience of cooperative abilities, such as increased precision of measurements with a stochastic element (e.g. localizing ionizing radiation sources [1]), distributing the application payload, and dynamic collaboration (e.g. cooperative object transport [2]).

Deployment of a single unmanned aerial vehicle (UAV) requires a complex system composed of several intricate subsystems handling the vehicle control, environment perception, absolute or relative localization, mapping, navigation, and communication. A

system scaled to a set of tightly cooperating UAVs must additionally introduce decentralized behavior generation, fault detection, information sharing in an often low-to-none bandwidth communication network, and detection and localization of inter-swarm members. Furthermore, the characteristic environments in the context of aerial swarms suited for real-world challenges may be unknown in advance, they incorporate high density of complex obstacles, they provide none-to-low access to mutual intercommunication between the team agents, and they allow either no access or unreliable access to a global navigation satellite system (GNSS). Each of these concepts is a complex challenge on its own. However, overcoming all the challenges opens the way to applications requiring distributed sensing and acting, such as cooperative area coverage for search & rescue, exploration, or surveillance tasks.

In this article, we present a complete swarm system framework, which respects the swarm and environment characteristics. The properties of the framework presented here correspond closely with the definition of autonomous swarms, as listed in [3]. The properties are: scalability for large groups, high redundancy and fault tolerance, usability in tasks unsolvable by a single robot, and locally limited sensing and communication abilities. Inspired by the self-organizing behavior of large swarms of homogeneous units with limited local information that is found among biological systems, our framework goes even further beyond the swarm requirements from [3] by dealing with all centralized and decentralized communication with the use of the ultraviolet direction and ranging (UVDAR) local perception method. The elimination of communication is particularly important in dense swarms of fast-moving aerial vehicles, where time-based delays in mutual localization might disturb the collective behavior of swarms and thus may induce mutual collisions. The independence from communication makes the system also applicable as a backup solution for swarm stabilization in scenarios where communication is required, but suffers from outages.

This allows us to employ a fully decentralized system architecture not limited by scalability constraints. This decentralization is advantageously robust towards a single-point of failure, reduces the hardware demands for individuals, and distributes the sensing and acting properties. We have been inspired mainly by ordinary representatives of biological systems: common starlings *Sturnus vulgaris*, which exhibit a remarkable ability to maintain cohesion as a group in highly uncertain environments and with limited, noisy information [4]. Similarly to starlings (and numerous other biological species), the proposed swarming system relies on sensing organs that look on two sides (cameras in our case), observing close-proximity neighbors only and responding to these sensory inputs by a local behavior which together forms a swarm intelligence that reaches beyond the abilities of a single particle.

The UVDAR method tackles the problem of mutual perception of swarm particles by localizing the bearing and the relative 3D position of their artificial ultraviolet (UV) light emission in time, using passive UV-sensitive cameras. The method is deployable in indoor and outdoor environments with no need for mutual communication or for a heavy-weight sensory setup. In addition, it is real-time, low-cost, scalable, and easy to plug into existing swarm systems. To verify the feasibility of the UVDAR technique in an aerial communication-less swarm system, we employed UVDAR to generate a decentralized bio-inspired swarming behavior employing local information about neighboring agents and close-proximity obstacles in real-world conditions. As verified in real-world experiments, the proposed system

for relative localization is accurate, robust, and reliable for use in decentralized local-information based swarming models (figure 1).

1.1. Related work

1.1.1. Relative localization

In most recent work concerning swarms and formation flight [5], the proposed algorithms have only been validated either in simulation or in laboratory-like conditions with the presence of absolute localization. This was merely converted to relative measurements virtually, using systems such as real-time kinematic (RTK)-GNSS or motion capture (mo-cap). It is well known that mo-cap is impractical for real-world deployment of mobile vehicles (either outdoors or indoors), as it requires the installation of an expensive infrastructure. These absolute localization sources can provide the full pose of tracked objects, which oversimplifies the whole task with respect to the reality of practical deployment. Even if only partial information derived from absolute measurements is passed to the UAVs (e.g. distance or bearing), the continuous stream of such information is produced without realistic errors, which is unrepresentative of real-world conditions.

Some more practical approaches consider infrastructure/less sensing such as ranging based on a radio signal [6]. This only allows for distance-based following, without any orientation information, and requires a specific motion for sufficient state observability. Another approach [7], for the 2D case, wirelessly communicates the intentions of the leader. This proves to be feasible since there are fewer degrees of freedom and there is less drift than in a general 3D case. These two approaches rely on radio transmission, which is subject to the effects of network congestion and interference. For this reason, we consider vision-based approaches more suitable for multi-robot groups, especially in uncontrolled outdoor environments.

This approach has previously been explored by the authors' research group, relying on true outdoor relative localization, see [8]. The source of the relative localization was an onboard vision-based system using passive circular markers, as described in [9]. There were, however, drawbacks: high sensitivity to the external lighting conditions and to partial occlusion, and substantial size for an acceptable detection range.

The use of active infrared (IR) markers has also been explored (see [10–12]) for the ability to suppress backgrounds using optical filtering. These methods are however suitable solely for indoor, laboratory-like conditions, since solar radiation excessively pollutes the IR spectrum, and subsequently the signal tends to deteriorate. In [12], the authors employed IR markers with blinking frequency in the kilohertz range, which required event-based cameras to detect micro-scale changes. These cameras are capable of detecting



Figure 1. A compact aerial swarm of three UAVs in a controlled outdoor environment filled with artificial obstacles, as viewed by an outside observer. The decentralized approach, described in detail in section 4, applies a set of local rules contributing to safe navigation and self-organization of the swarm structure among obstacles. The UAVs are homogeneous units with solely local sensing.

micro-scale changes. However, they typically do not provide sufficiently high field of view and resolution, and they are not suitable for scalable swarms due to their size and cost. The IR spectrum has also been utilized in a passive manner [13], but this approach, though simple, is even less robust to the outdoor conditions and distances applicable to UAVs.

It is also feasible to visually detect and localize unmarked UAVs using machine learning (ML) methods such as convolutional neural networks. However, these approaches require meticulously annotated datasets with a specific UAV and with an environment similar to the intended operational space [14, 15]. The computational complexity and the dependency on satisfactory lighting conditions of such ML systems precludes their deployment onboard lightweight UAVs suitable for swarming. This motivated the development of the UVDAR system, which is more robust to real-world conditions, because it reduces the computational load by optically filtering out visual information that is not of interest. In contrast to [14, 15], UVDAR also provides target identities. The whole sensor is small, lightweight, and does not depend on the external lighting conditions.

1.1.2. System architecture

To date, deployments of real-world aerial teams have not used any of the methodologies of direct localization described here in order to deal with the mesh-communication between the team members or with the communication link with a centralization element. The record in terms of the number of UAVs cooperating at the same time is currently held by Intel[®] [16] with its fleet of shooting star quad-rotors. Intel's centralized solution performs spectacular artistic light shows. However in Intel's arrangement, each team member follows a pre-programmed trajectory, relying on GNSS and a

communication link with a ground station. A similar methodology is employed in [17–19], where the authors deployed swarms of UAVs in order to verify bio-inspired flocking behaviors in known confined environments. In comparison with [16], their methods are decentralized; however, the UAVs still communicate their global states obtained by GNSS within a radio-frequency mesh network. This is not a realistic assumption in most application scenarios.

Recent successful real-world deployments are summarized in table 1. Observe that some kind of communication (either ground station to unit or unit-to-unit) is employed in most of the related work. The dependency on a communication network lowers the upper limit for swarm scalability, due to the bandwidth limitations, and significantly reduces the fault tolerance of the entire system. The UVDAR relative visual perception system, described in detail in section 3, is designed to remove this dependency. Its use may allow working swarm systems to mimic the local behavioral mechanisms found in biological systems, ranging from general flocking to leader-follower scenarios.

1.1.3. Swarm stabilization

To enable short-term stabilization of an autonomous UAV, an onboard inertial measurement unit (IMU) directly measures its linear acceleration, the attitude and the angular rate, using a combination of accelerometers, gyroscopes, and magnetometers. To obtain long-term stabilization of an UAV, however, it is not sufficient to use only the onboard IMU, due to the inevitable measurement noises and drifts. It is common practice to provide an additional estimate of the state vector variables (typically position or velocity), which is fused together with all the inertial measurements. The most common approach is to estimate the global position using a GNSS. However,

Table 1. A brief comparison of aerial swarm systems with successful recent deployments outside of laboratory-like conditions. Methods marked with (*) employ communication with a centralized ground station.

Work	Decentralized	Communication	Relative localization
Intel® [16]	No	Yes*	Shared global position (WiFi)
EHang, Inc. [20]	No	Yes*	Shared global position (WiFi)
Hauert et al [21]	Yes	Yes	Shared global position (WiFi)
Bürkle et al [22]	Yes	Yes	Shared global position (WiFi)
Kushleyev et al [23]	No	Yes*	Shared global position (ZigBee)
Vásárhelyi et al [17–19]	Yes	Yes	Shared global position (XBee)
Weinstein et al [24]	No	Yes*	Shared global position (WiFi)
Stirling et al [25]	Yes	Yes	IR ranging
Nguyen et al [6]	N/A	Yes	Ultra-wideband ranging
Nägeli et al [26]	Yes	Yes	Visual markers
This work	Yes	No	UVDAR

GNSS signal availability is limited strictly to outdoor environments, and the accuracy of GNSS is affected by an error of up to 5 m [27]. Although the accuracy can be improved to 2 cm with the use of RTK-GNSS, this makes aerial swarms deployable solely in controlled environments and is in contradiction with the bio-mimicking premise, since precise global localization is uncommon in biological systems. Other common methods of state estimation are local, and they typically employ onboard laser- or vision-based sensors to produce local estimates of the state variables. Vision-based methods may compute the optical flow to estimate the velocity of the camera relative to the projected image plane [28], or may apply algorithms of simultaneous localization and mapping to visual data [29]. Laser-based sensors are mostly used to estimate the relative motion between two frames of generated point-cloud data [30].

There are structurally two approaches for stabilizing a swarm in a decentralized manner. The first group of methods distributes the state estimates determined for individual self-stabilization throughout the swarm (see table 1). In addition to restricting the communication infrastructure, this methodology has a major dependency between the swarm density and the accuracy of the global localization (e.g. GNSS). In addition, it requires knowledge of individual transformations amid the coordination frames for distributed local state estimation methods. The second group of methods does not adopt a communication network to distribute the state estimates, but rather estimates the states directly from the relative onboard observations. This approach makes the swarm independent from the infrastructure, but it makes direct detection, estimation, and decision making with limited information more challenging. As further shown in section 5, the developed framework is part of the second group, perceiving the local neighborhood with visual organs and deploying a swarm of UAVs in fully-decentralized manner.

1.1.4. Swarming without communication

Decentralized swarming models accounting for complete or partial absence of communication were explored exclusively for 2D systems in the past (this is also implied in table 1). The majority of the state-of-the-art works within this field are biologically-inspired and emphasize self-organizing behavior of large-scale swarms of simple units with highly limited sensory capabilities. Highlighted is the Beeclust [31] approach, which uses probabilistic finite state machines and a primitive motion model to mimic the collective behavior of honeybees. The Beeclust can be applied to complex tasks where information exchange among units is not required, such as in underwater exploration using a swarm of underwater robots [32]. A different method [33] analyzes the aggregation of agents towards a common spatial goal while avoiding inter-agent collisions. The authors of [33] show that their method with limited sensing properties of the agents performs similarly to methods employing complete pose information. All of these decentralized algorithms require some form of mutual relative localization (even limited to binary detections), making them suitable for the use of UVDAR localization. Overall review of the 2D approaches is systematically described in [34], which further highlights the lack of research focus in the field of aerial swarming in 3D space.

1.2. Contributions

This article addresses problems of the deployment of real-world aerial swarms with no allowed communication or position sharing. This potential problem is overcome with the use of the novel vision-based UVDAR system for direct mutual perception of team members. The stability of the UVDAR system for use in aerial swarming is the outcome of thorough real-world experimental verification in an outdoor environment with and without obstacles. The main features of this article are as follows:

- (a) It provides an enabling technology for swarm research, often bio-inspired, by introducing a system that achieves fundamental swarm properties, as defined in [3].
- (b) It introduces the UVDAR system as an off-the-shelf tool for relative localization and identification of teammates suited for mutual perception of agents in robotic systems, such as aerial swarms.
- (c) It introduces a decentralized bio-inspired swarming approach suited for obstacle-filled real-world environments, which requires only local relative information and no mutual communication.
- (d) It verifies the feasibility and analyses the usability of aerial flocking relying on direct localization, which is the most frequent mechanism in biological systems.
- (e) It is based on several real-world deployments of aerial swarms.
- (f) It presents, to the best of our knowledge, the first autonomous deployments of aerial swarms with no centralized element and no mutual communication.
- (g) It discloses the entire system as open source at <https://github.com/ctu-mrs>.

2. Motivation

The lack of a communication-independent approach has put a constraint on much of the work done until now in the field of deploying teams of unmanned vehicles in challenging environments. Our work here is motivated by the need for a communication-independent approach, and presents solutions that we have developed. The insights into the development of the real-world deployments presented here tackle the motivations and constraints of the vast majority of related work restrained by the heretofore lack of communication-independent approaches.

Focusing on dense swarms of UAVs with short mutual distances, most of the swarming approaches reported in the literature have not been tested in real-world conditions. Theoretical derivations, software simulations, and occasional experiments in laboratory conditions have formed the target for most of the related literature, as analyzed in [5, 35]. However, this research milestone is far away from a meaningful real-world verification needed for an applicability of aerial swarms. Real world interference cannot be neglected, as the integration of a swarming intelligence onto a multi-robot system yields constraints that need to be characterized directly in models of swarming behavior.

Instigated by biologically-inspired swarming models [35, 36] capable of achieving complex tasks (e.g. navigation, cohesion, food scouting, nest guarding, and predator avoidance) with a team of simple

units, our aim was to imitate these models with the use of local information, as is widely observed in nature. To allow the deployment of an infrastructure-independent (communication, environment) model, we had identified the most crucial factor impeding this type of deployment of a decentralized architecture—the mutual relative localization between team members, which is also the most crucial information for animals in flocks in nature. This motivated the development of the UVDAR system (see section 3), designed as a light-weight off-the-shelf plugin providing the local localization of neighboring swarm particles. The usability of UVDAR in dense swarms is analyzed in detail in section 6.

3. UVDAR

Inspired by our extensive prior experimental experience with vision-based relative localization of UAVs (see [9, 37]), we developed a novel relative localization sensor that tackles various limitations of previous solutions, namely the unpredictability of outdoor lighting and limits on the size and weight of onboard equipment. The sensor, named UVDAR, is a UV vision-based system comprising a UV-sensitive camera and active UV light-emitting diode (LED) markers. These lightweight, unobtrusive markers, attached to extreme points of a target UAV, are seen as unique bright points in the UV camera image (see figure 2). This allows computationally simple detection [38] and yields directly the relative bearing information of each marker from the perspective of the camera. The fish-eye lenses that are used with the UV camera provide a 180° horizontal overview of the surroundings. Known camera calibration, together with the geometrical layout of the markers on the target, allows us also to retrieve an estimate of the distance (see [38, 39] for details).

In order to provide specific markers that would be distinguishable from others, and also to provide a further increase in robustness with respect to outliers, we set the markers to blink with a specific sequence. Using our specialized implementation of the 3D time-position Hough transform (see [38] for details), we can retrieve this signal for each observed marker, giving them identities. In this project, we use these IDs to simplify the separation of multiple observed neighbor UAVs, but they can also be used to retrieve the relative orientation of the neighbors [39]. In addition to the swarming application described in this paper, UVDAR may be used for e.g. a directed leader-follower flight [39], where the use of the retrieved orientation is essential. In addition, the neighbors' orientation estimate can be exploited for automatic generation of a dataset for training ML vision for UAV detection, as applied in [40], where UVDAR was used for annotating color camera images.

In swarms and in multi-UAV systems in general, the blinking frequency of the onboard LEDs can be

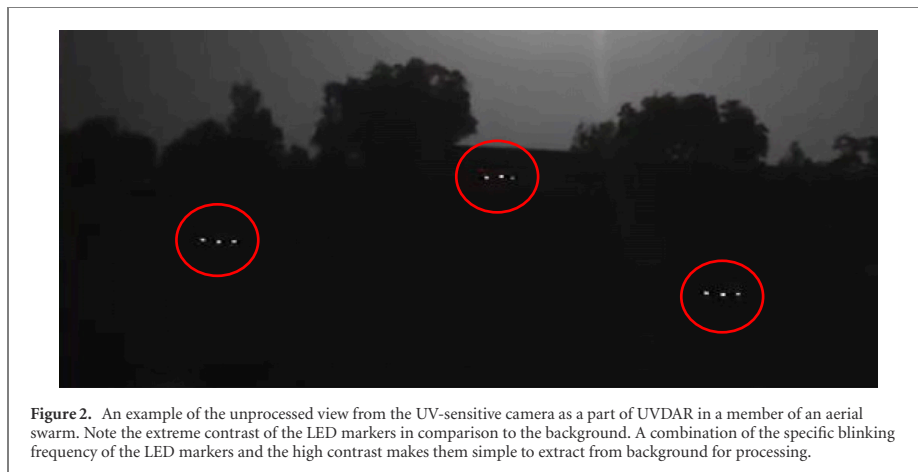


Figure 2. An example of the unprocessed view from the UV-sensitive camera as a part of UVDAR in a member of an aerial swarm. Note the extreme contrast of the LED markers in comparison to the background. A combination of the specific blinking frequency of the LED markers and the high contrast makes them simple to extract from background for processing.

configured to encode information for optical data transmission between swarm units, in addition to using LED blinking directly for relative localization. An example of such an application is in exploration, where a scouting unit can indicate the presence and the relative position of a discovered target to other units by combining various blinking signals and the unit's own orientation. A further use is in cooperative voting in a group, where each unit expresses the current selection with blinking signals, and adjusts its vote on the basis of observing the selections of others.

In this paper, we go beyond our preliminary works with UVDAR [38–40], and also beyond other state-of-the-art literature, by incorporating direct mutual localization of UAVs into the position control feedback loop of a fully-decentralized swarming system without any kind of communication and external localization. To the best of our knowledge, this paper presents the first real-world deployments of fully-decentralized bio-inspired swarms of UAVs using direct local localization for collective navigation in an uncontrolled environment. This is what UVDAR was intended for.

3.1. Safety

The use of UV radiation in the system has understandably raised some health concerns in the past. We have verified the safety of this application by consulting the International Commission on Non-Ionizing Radiation Protection (ICNIRP) 'guidelines on limits of exposure to UV radiation of wavelengths between 180 nm and 400 nm' [41]. According to these guidelines, the exposure to UV radiation (both to the eyes and to the skin) should not exceed 30 J m^{-2} weighted by the relative spectral effectiveness (unitless wavelength-specific factor). In our case of 395 nm radiation, this factor equals to 0.000 036, making the actual limit $8.3 \times 10^5 \text{ J m}^{-2}$. This means that our LEDs, producing 230 mW of total radiated power [42] at the given driving current, can be safely viewed

from the distance of 1 m from the frontal direction (with the highest intensity in its Lambertian radiation pattern) for over 3000 h, making it effectively harmless.

3.2. Scalability

In the context of a robotic swarm, scalability of the whole system is an important factor. Using a communication network in large groups of robots limits the scalability by an upper bound defined by the total bandwidth, by the number of available channels, by the network architecture, or by the required data flow. Employing a local perception method such as UVDAR, the state of *swarm particles* (team members, swarm units) is shared via direct observations, as is common in swarms in nature. This system therefore does not need an explicit radio communication network.

As a vision-based method, UVDAR suffers from natural restrictions, namely visual occlusions, camera resolution, and the detection, separation, and identification of image objects. The upper scalability bound is determined by the ability to filter out the UV markers belonging to a given swarm agent. If the markers of all UAVs in the swarm are set to blink with the same frequency, individual agents have to be distinguished by separating their positions in the UV image and in the constellations that they form. In this case, we estimate that each agent should be capable of distinguishing up to 30 neighboring agents within the range of the UVDAR system, bounded by the computational limitations. This is however not the ideal mode of operation, as it becomes problematic when there are occlusions between agents, or when the agents are in close proximity in the observed image.

To tackle this challenge, we apply different blinking frequencies to different agents. The UVDAR system in its current configuration can accommodate up to six different frequencies of blinking that can be reliably distinguished from each other. This allows us

to mitigate the issue of overlapping agents—indeed, even agents that are directly behind each other can often be separated, if extreme markers of the further agent protrude into the image. However, since the number of usable blinking frequencies is limited, we need to devise a method for spreading them evenly in the swarm, such that the likelihood of image separation of overlapping agents based on different frequencies between them is maximized for the whole swarm. This has to be done in a decentralized manner, in order not to violate the swarming paradigm.

One way to solve this for dense UAV swarms is to have each agent dynamically re-assign its blinking frequency to differ as much as possible from the neighbors that it observes. This challenge definition can be likewise defined as the constraint satisfaction problem solved within a decentralized swarm of UAVs using direct observations only. The idea of this method is to maximize the local frequency diversity and additionally to allow all of the agents to initiate with the same ID (encoded by the blinking frequency of onboard markers). This opposes the current methodology of manually pre-setting the frequencies before deployment (see section 6). The analysis and the theoretical limits on the convergence of such an approach towards a stable final state maximizing the scalability bound is still underway.

Another approach to increase the scalability bound, while carrying the identical ID on all the agents, lies in the design of UVDAR itself. It is possible to introduce an additional omnidirectional UV source on top of each agent. This additional source is called a *beacon* and it blinks with a specific frequency unique to the rest of the onboard markers on an agent. This allows for the separation of pixels in the image stream based on their image distance as well as their association with the singular beacon marker. The presence of at least two beacons in one region of the observer's image clearly implies a partial mutual occlusion. The use of beacons hence provides a limited ability to separate even agents in partial mutual occlusion relative to an observer if the beacons of both agents are visible.

The maximum range of detection should be taken into account for scalability in the geometrical sense. With the current UVDAR setup, detection is possible for targets up to 15 m away from the sensor. However, for improved reliability and robustness, a maximum range of 10 m is recommended. For determining the theoretical accuracy and range limitations, see [38]. For a quantitative analysis on real-world accuracy, see section 6.3. Filtering out distant targets, the limited detection range makes the method suitable for dense swarms, which place emphasis on a number of entities in a local neighborhood rather than on the swarm as a whole. In biological systems, this perception characteristic allows for swarms of utmost magnitude, such as fish schools [43] with thousands of entities.

4. Swarming intelligence

In this article, we follow the swarm concept defined in section 1, in which the group is composed of swarm units with limited computational power and a short-term memory. The concept is decentralized and uses autonomous self-organizing groups of homogeneous aerial vehicles operating in a 3D space.

The proposed flocking approach works entirely with local information, with no requirement for any form of radio communication between the homogeneous swarm particles, and in an environment with convex obstacles. The approach is inspired by biological systems, where global cooperative behavior can be found to emerge from elementary local interactions. We will show that this phenomenon of cooperative behavior may yield collision-free stabilization in cluttered environments, self-organization of the swarm structure, and an ability to navigate in tasks suited for real UAVs. The proposed swarming framework is founded on previously developed models [44, 45], which have been enhanced to suit the demands of real-world interference by extending them with concepts of obstacle avoidance, perception, and navigation. The introduction of such extension concepts is highly important as the assumptions of dimensionless particles and an ideal world as in [44, 45] do not apply in the real world. The main idea of the swarming behavior presented here is to verify the feasibility, to perform an analysis, and to derive the properties of the UVDAR system for use in swarm systems. Bear in mind that UVDAR is a general system and any swarming model [17, 33, 46], formation control approach [47], or obstacle/predator avoidance method [48] utilizing local relative information can be employed to generate intelligent behavior when employing the UVDAR system.

4.1. Behavior generation

The behavioral model used throughout this article is defined in discrete time step k for a homogeneous swarm unit i with an observation radius $R_n^i \in \mathbb{R}^{>0}$, an obstacle detection radius $R_o^i \in \mathbb{R}^{>0}$, a swarming velocity $\mathbf{v}_{[k]}^i \in \mathbb{R}^{3 \times 1}$, and a set of locally detected neighbors $\mathcal{N}_{[k]}^i$ within the observation radius R_n^i , as follows. Bear in mind that all the relative observations in particle i are given in the body frame of particle i at time step k .

The individual detected neighbor particles $j \in \mathcal{N}_{[k]}^i$ are represented by vectors of relative position $\mathbf{x}_{[k]}^{ij} \in \mathbb{R}^{3 \times 1}$ and relative velocity $\mathbf{v}_{[k]}^{ij} \in \mathbb{R}^{3 \times 1}$, $\forall j \in \{1, \dots, |\mathcal{N}_{[k]}^i|\}$, defined as

$$\mathbf{x}_{[k]}^{ij} = [\mathbf{x}_{[k]}^{ij}, \mathbf{y}_{[k]}^{ij}, \mathbf{z}_{[k]}^{ij}]^T, \quad (1)$$

$$\mathbf{v}_{[k]}^{ij} = \frac{1}{\Delta t_{[k]}^{ij}} (\mathbf{x}_{[k]}^{ij} - \mathbf{x}_{[k-1]}^{ij}) - \mathbf{v}_{[k-1]}^i, \quad (2)$$

where $x_{[k]}^{ij}, y_{[k]}^{ij}, z_{[k]}^{ij}$ are Cartesian coordinates of a neighbor particle j represented in the body frame of agent i in time step k , $\Delta t_{[k]}^{ij} = t_{[k]}^{ij} - t_{[k-1]}^{ij}$ is the time elapsed since the last direct detection of neighbor j , and $\mathbf{v}_{[k=0]}^i = \mathbf{v}_{[k=0]}^j = \mathbf{0}$. The swarming model is then defined as a sum of elementary forces

$$\mathbf{f}_{[k]}^i(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i) = \mathbf{f}_{[k]}^{b,i}(\mathcal{N}_{[k]}^i) + \mathbf{f}_{[k]}^{n,i}(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i), \quad (3)$$

where $\mathbf{f}_{[k]}^{b,i}(\cdot) \in \mathbb{R}^{3 \times 1}$ embodies the baseline forces as an interpretation of the *Boids model* [44] flocking rules *cohesion*, *alignment*, and *separation*, modified for real UAVs as

$$\begin{aligned} \mathbf{f}_{[k]}^{b,i}(\mathcal{N}_{[k]}^i) &= \frac{1}{|\mathcal{N}_{[k]}^i|} \sum_{j=1}^{|\mathcal{N}_{[k]}^i|} \\ &\times \left[\mathbf{x}_{[k]}^{ij} + \frac{\mathbf{v}_{[k]}^{ij}}{\lambda} - \kappa(\mathbf{x}_{[k]}^{ij}, R_n^i) \mathbf{x}_{[k]}^{ij} \right]. \end{aligned} \quad (4)$$

The scalar λ (Hz) is the update rate of direct localization (camera rate) and the weighting function

$$\kappa(\mathbf{x}, r) = \max \left(0; \frac{\sqrt{\|\mathbf{x}\|_2} - \sqrt{r}}{\|\mathbf{x}\|_2} \right) \quad (5)$$

represents a nonlinear weight coefficient scaling the repulsion behavior by the mutual distance between two neighbors. As the original model [44] was designed for swarms of dimensionless particles, function $\kappa(\cdot)$ is particularly important for a swarm of real UAVs, in order to prevent mutual collisions while maintaining flexibility of the swarm as a whole. The force $\mathbf{f}_{[k]}^{n,i}(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i) \in \mathbb{R}^{3 \times 1}$ in (3) is an extension to the simple model [44] in the form of an additional navigation rule in an environment composed of $\mathcal{N}_{[k]}^i$ and a set of obstacles $\mathcal{O}_{[k]}^i$ detected within the detection radius R_o^i .

The navigation rule can exploit any local multi-robot planning method [49–51] in order to optimize the swarm motion parameters and to prevent a deadlock situation, or can include an obstacle avoidance mechanism and a navigation mechanism by introducing them as additional simplistic rules. To provide an example of the system performance, we introduce a simple attraction force $\mathbf{v}_{[k]}^{n,i} \in \mathbb{R}^{3 \times 1}$ towards a specified goal, together with a local reactive obstacle avoidance rule. To represent the obstacles, we introduce the concept of a *virtual swarm particle*, which efficiently replaces a general geometric obstacle by a virtual entity. This dimensionless particle is represented by a state comprised of a position and velocity relative to particle i , similarly as defined in (1) and (2). The methodology for finding the state of a virtual swarm particle is derived in the following section. The navigation rule is then derived as

$$\begin{aligned} \mathbf{f}_{[k]}^{n,i}(\mathcal{O}_{[k]}^i) &= \frac{1}{|\mathcal{O}_{[k]}^i|} \sum_{v=1}^{|\mathcal{O}_{[k]}^i|} \\ &\times \left[\frac{\mathbf{v}_{[k]}^{iv}}{\lambda} - \kappa(\mathbf{x}_{[k]}^{iv}, R_o^i) \mathbf{x}_{[k]}^{iv} \right] + \frac{\mathbf{v}_{[k]}^{n,i}}{\lambda}, \end{aligned} \quad (6)$$

where the vectors of the relative position $\mathbf{x}_{[k]}^{iv} \in \mathbb{R}^{3 \times 1}$ and the relative velocity $\mathbf{v}_{[k]}^{iv} \in \mathbb{R}^{3 \times 1}$ constitute the state of a v th virtual swarm particle.

The swarming model defined in (3) represents the steering force of a particle i , which is used to compute the swarming velocity of particle i as

$$\mathbf{v}_{[k]}^i = \gamma(\mathbf{f}_{[k]}^i(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i)) \frac{\mathbf{f}_{[k]}^i(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i)}{\|\mathbf{f}_{[k]}^i(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i)\|_2}, \quad (7)$$

where

$$\gamma(\mathbf{f}) = \min \{v_m; \lambda \|\mathbf{f}\|_2\} \quad (8)$$

bounds the magnitude of the velocity below the maximum allowed speed v_m (m s^{-1}). The swarming velocity is then used in real-world applications to compute the desired position setpoint as

$$\mathbf{r}_{[k]}^{d,i} = \frac{\mathbf{v}_{[k]}^i}{\lambda} \quad (9)$$

represented in the body frame of UAV i .

4.2. Obstacle detection

To achieve flocking in the targeted environment (e.g. a forest environment and an indoor environment), the obstacles in the local neighborhood are generalized into two geometrical classes (circles and lines), based on their cross-sections with the horizontal plane of a particle, as portrayed in figure 3. This assumption allows us to model more complex settings (e.g. a forest or an office-like environment) on the grounds of these two geometrical classes, while it throttles down the perception and the computational complexity onboard a lightweight UAV. Detection of these obstacles is assumed to be provided for a particle i from any kind of an onboard sensor with an obstacle detection distance R_o^i .

Having in time step k a detected circular obstacle v with a radius $r_{[k]}^v \in \mathbb{R}^{>0}$ and a center at $\mathbf{c}_{[k]}^{iv} \in \mathbb{R}^{3 \times 1}$ referenced in the body frame of particle i , the state of a v th virtual swarm particle is derived as

$$\mathbf{x}_{[k]}^{iv} = \left(1 - \frac{r_{[k]}^v}{\|\mathbf{c}_{[k]}^{iv}\|_2} \right) \mathbf{c}_{[k]}^{iv}, \quad (10)$$

$$\mathbf{v}_{[k]}^{iv} = \frac{r_{[k]}^v}{\|\mathbf{c}_{[k]}^{iv}\|_2} \left(\mathbf{I} - \boldsymbol{\mu}_{[k]}^{iv} (\boldsymbol{\mu}_{[k]}^{iv})^T \right) \mathbf{v}_{[k]}^i, \quad (11)$$

where $\|\cdot\|_2$ is the L^2 norm, $\mathbf{I} \in \mathbb{R}^{3 \times 3}$ is an identity matrix, and $\boldsymbol{\mu}_{[k]}^{iv} = \frac{\mathbf{c}_{[k]}^{iv}}{\|\mathbf{c}_{[k]}^{iv}\|_2}$. By analogy, the virtual swarm agent state can be derived for a linear obstacle

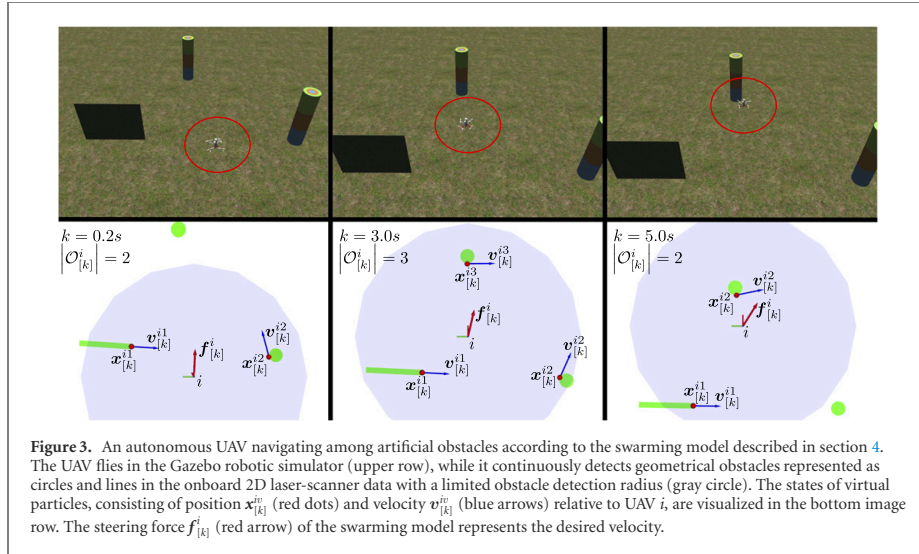


Figure 3. An autonomous UAV navigating among artificial obstacles according to the swarming model described in section 4. The UAV flies in the Gazebo robotic simulator (upper row), while it continuously detects geometrical obstacles represented as circles and lines in the onboard 2D laser-scanner data with a limited obstacle detection radius (gray circle). The states of virtual particles, consisting of position $\mathbf{x}_{[k]}^i$ (red dots) and velocity $\mathbf{v}_{[k]}^i$ (blue arrows) relative to UAV i , are visualized in the bottom image row. The steering force $\mathbf{f}_{[k]}^i$ (red arrow) of the swarming model represents the desired velocity.

defined by its normal vector $\mathbf{n}_{[k]}^{iv} \in \mathbb{R}^{3 \times 1}$ and a set of observed points $\mathcal{P}_{[k]}^{iv}$ as

$$\mathbf{x}_{[k]}^{iv} = (\mathbf{I} - \mathbf{P}_{[k]}^{iv}) \hat{\mathbf{p}}_{[k]}^{iv} \quad (12)$$

$$\mathbf{v}_{[k]}^{iv} = \frac{1}{\|\hat{\mathbf{p}}_{[k]}^{iv}\|_2} \mathbf{P}_{[k]}^{iv} \mathbf{v}_{[k]}^i, \quad (13)$$

where

$$\mathbf{P}_{[k]}^{iv} = \mathbf{I} - \mathbf{n}_{[k]}^{iv} (\mathbf{n}_{[k]}^{iv})^T, \quad (14)$$

$$\hat{\mathbf{p}}_{[k]}^{iv} = \arg \min_{\mathbf{p} \in \mathcal{P}_{[k]}^{iv}} \{\|\mathbf{p}\|_2\}. \quad (15)$$

The state of a virtual swarm particle for both geometrical classes is visualized in figure 3, where an autonomous UAV navigates among artificial obstacles within an environment of the Gazebo robotic simulator.

5. System architecture

In addition to the method for direct onboard localization presented in section 3 and the decentralized swarming approach presented in section 4, we will now present here system architecture of the entire UAV system, supplemented by the concepts of UAV stabilization, control, and state estimation. These concepts are based on our previous research (see [1, 37, 52]) focused on cooperation among autonomous aerial vehicles. They have been adapted for swarming research described in this article. The control pipeline, suited for stabilizing and controlling UAV swarms using linear model predictive control (MPC) and the non-linear $\text{SO}(3)$ state feedback controller [53], is depicted in the high-level scheme in figure 4. The stabilization and control pipeline is based entirely on [52].

In addition, a decentralized collision avoidance system [55] is adapted in the proposed system for safe research on compact aerial swarms. A long prediction horizon of linear MPC is used to detect collisions among trajectories of robots. The known collision trajectories are then altered prior their execution. This allows us to implement the collision avoidance system in a decentralized manner. Decentralized collision avoidance is necessary for safe verification of bio-inspired swarming models in the real world. Although the use of mutual communication for collision avoidance is in contradiction with the system architecture presented in this article, it can be used as a low-level safety supervisor with no direct dependency on the architecture of the tested swarming model. This may prevent inadmissible collisions when there is undesired demeanor of dense swarm members, and therefore protect the hardware during the initial phases of experimental swarm deployment. However, the use of collision avoidance is not mandatory and its use is appropriate only during the initial testing phase.

To stabilize UAVs using the system in figure 4, the individual UAVs estimate their state vector

$$\mathbf{x} = [\mathbf{r}, \dot{\mathbf{r}}, \ddot{\mathbf{r}}, \mathbf{R}, \boldsymbol{\omega}]^T, \quad (16)$$

where $\mathbf{R} \in \text{SO}(3)$ is the attitude and $\mathbf{r} = [x_w, y_w, z_w]^T$ is the position in the world coordinate frame. The vector $\dot{\mathbf{r}} \in \mathbb{R}^{3 \times 1}$ is the linear velocity, $\ddot{\mathbf{r}} \in \mathbb{R}^{3 \times 1}$ is the linear acceleration, and $\boldsymbol{\omega} \in \mathbb{R}^{3 \times 1}$ is the angular rate with respect to the UAV body coordinate frame. The PixHawk autopilot [56] is embedded to handle the low-level attitude rate and actuator control, and an IMU is used to directly measure the linear acceleration $\ddot{\mathbf{r}}$, the attitude \mathbf{R} , and the angular rate $\boldsymbol{\omega}$, using a combination of accelerometers, gyroscopes, and magnetometers. The embedded autopilot integrates the

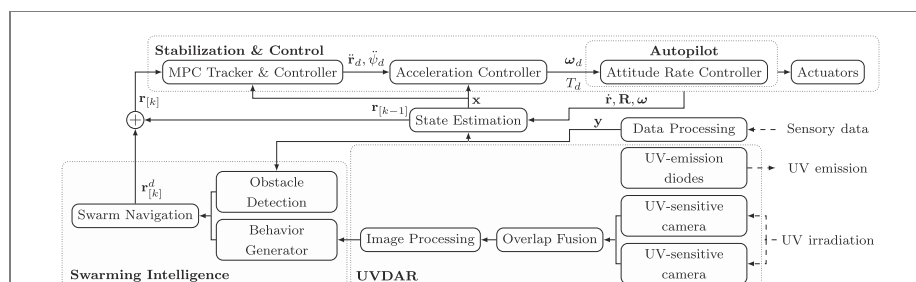


Figure 4. The high-level system pipeline (the schematic is based on the system pipeline diagram published in [54]) of a single homogeneous UAV swarm unit i in time step k . The stabilization & control pipeline [52] takes reference position setpoint $r_{[k]}$ for the MPC in the MPC tracker, which outputs a command $\ddot{r}_d, \ddot{\psi}_d$ ($\ddot{\psi}$ is the heading acceleration) for the acceleration tracking SO(3) controller [53]. The acceleration controller produces the desired angular rate ω_d and thrust reference T_d for the embedded attitude rate controller. A state estimation pipeline outputs the current state estimate x based on the sensory data y and the onboard measurements of linear velocity \dot{r} , angular rate ω , and attitude R . Note that the time indices of the stabilization & control and the state estimation pipelines are omitted in the diagram, since their timeline matches the rate of the inertial measurements (typically 100 Hz), which differs from the timeline of the detection cameras (10–20 Hz). Local perception of neighboring units using the UVDAR sensor is described in detail in section 3, while the decentralized swarming approach is described thoroughly in section 4.

measurements of \dot{r} to \dot{r} and employs the extended Kalman filter to produce optimal estimates of the specific state variables with respect to the measurement noise.

To self-localize an individual UAV, its global position measured by GNSS is fused together with the inertial measurements in order to stabilize the flight of this dynamically unstable system. However, the global state is not shared to other swarm agents throughout our final experimental analysis presented in section 6. Instead, the framework uses UVDAR to directly observe the relative position and the relative velocity (see (1) and (2)) of particles in the local neighborhood, and it generates a navigation decision based on the set of simple rules described in section 4. Although the use of GNSS for self-localization limits the system exclusively to outdoor environments, this dependency can be replaced by any local state estimation method with respect to the desired application and environment—e.g. the deployment of our decentralized system in a real-world forest, which was highlighted by the IEEE spectrum².

5.1. Properties

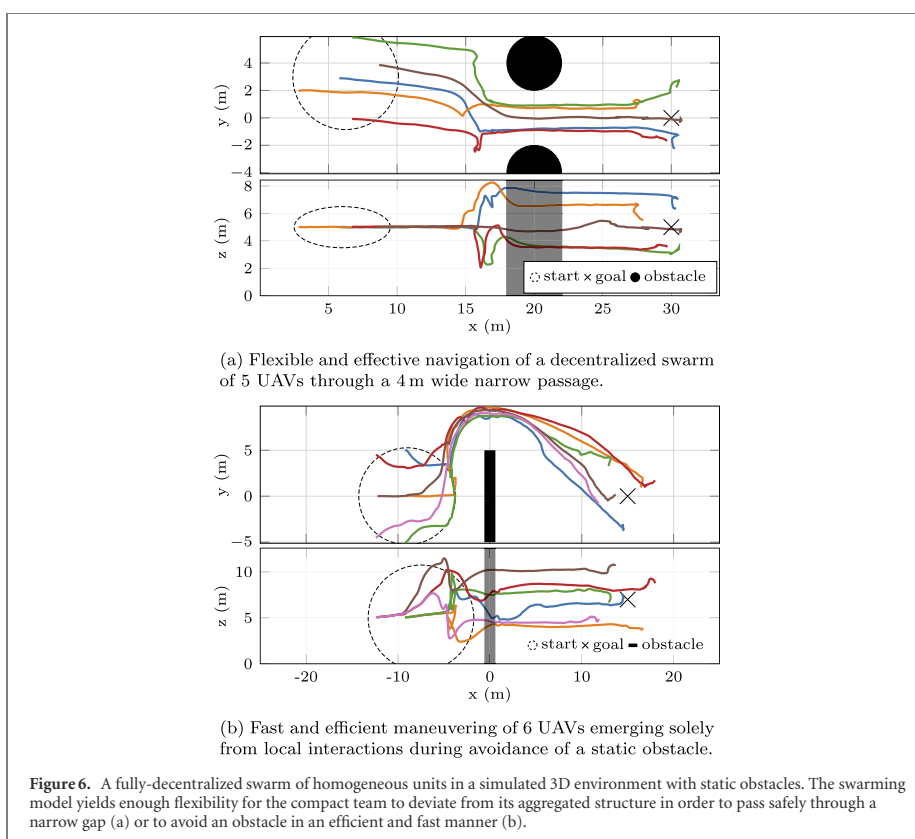
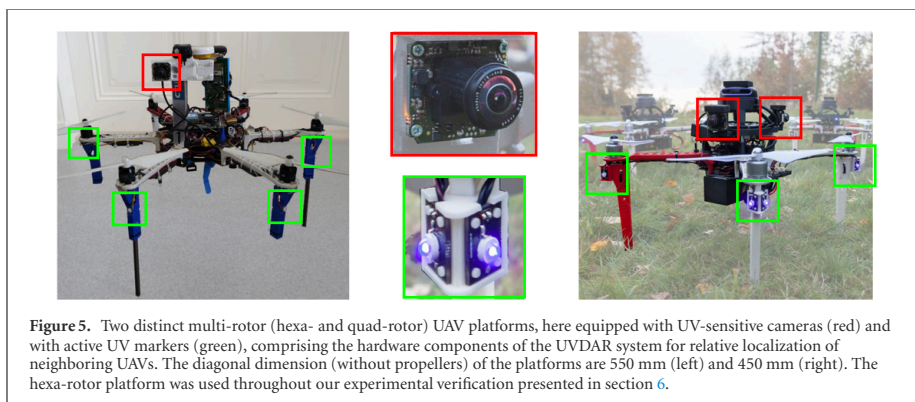
The combination of the system decentralization and the local perception of individual agents makes the system as a whole robust towards failures of individuals. In the swarming model (see section 4), each agent decides on its actions in real time only from current observations or a short-past history of observations. This makes the system robust towards a single-point of failure, such as a failure of some centralized control element or the communication infrastructure. Unless the employed local perception method generates false negative detections, the swarming model (see section 4) ensures no mutual collisions between

the agents. The rate of false negative detections in UVDAR is minimal as there are no objects blinking at specific rates in the given near-visible UV spectrum. In case of a hardware failure of an aerial agent (e.g. the agent lands unexpectedly), the agent disappears from the visibility field of other units resulting in emergent self-organization of the collective configuration.

As UVDAR is a vision-based system, it naturally suffers from visual occlusions generating blind spots in overcrowded situations. As discussed in section 3.2, the number of visual occlusions in UVDAR is mitigated with the use of different blinking frequencies of overlapping UAVs. As the neighborhood for perception is also locally limited in the swarming model (see section 4), the distant blind spots are filtered out in principle. The remaining occluded agents are neglected. This is feasible in the employed model, as the information about the units' presence is propagated through direct observations of the motion of the middle agents (i.e. the agents causing the occlusions). Based on our empirical experience, this does not destabilize the swarm, but rather rearranges the agents to positions where the number of visual occlusions is reduced.

The navigational features of the system as a whole are controlled in a decentralized manner. A decentralized navigation is possible with a swarming model capable of navigational decision making using only the perceived data onboard the units. This is the case of our swarming model (see section 4), which employs a simple steering towards a pre-specified set of global positions, hence eliminating the need for navigation managed by a centralized controller. Although our later experiments (see section 6) navigate each UAV individually, the model may navigate only a single unit with the rest of the swarm naturally following the leader—a behavior emerging from the cohesion and the alignment premises.

² <https://spectrum.ieee.org/automaton/robotics/drones/video-friday-dji-mavic-mini-palm-sized-foldable-drone>.

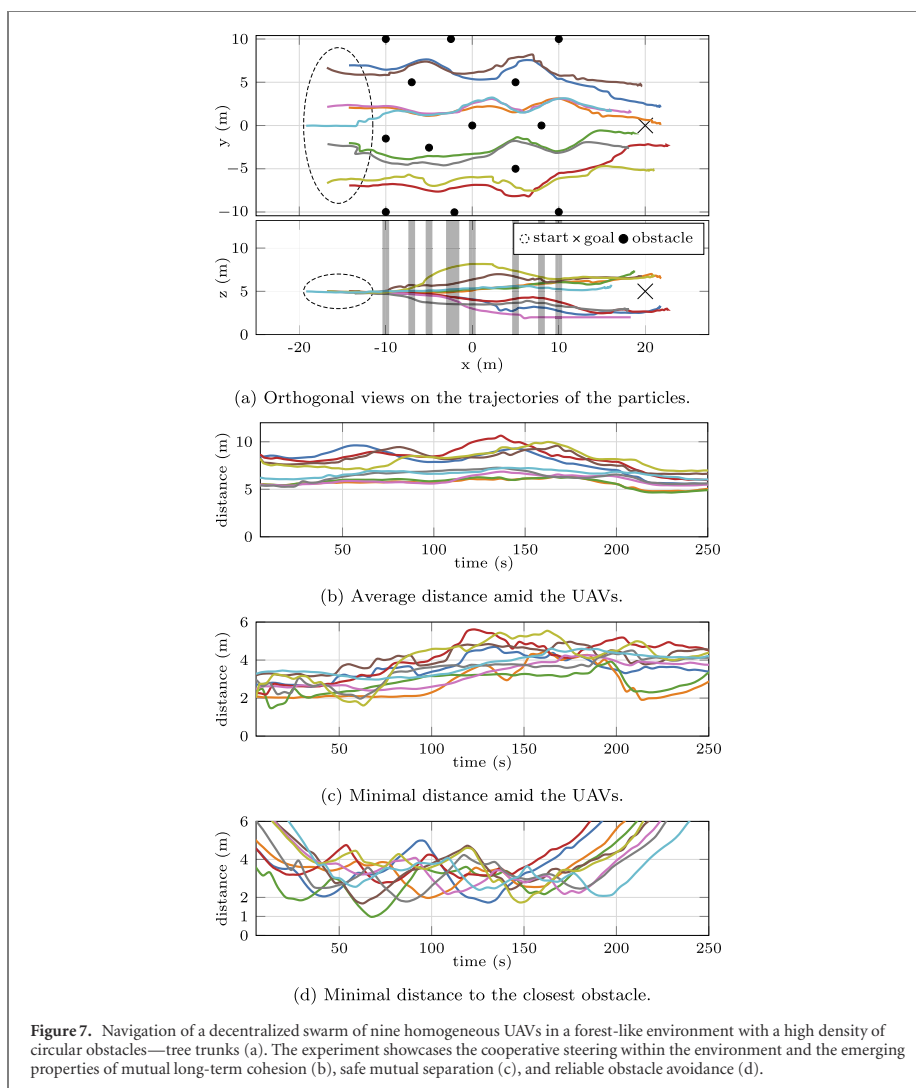


5.2. Hardware platform

The use of UVDAR is not dependent on the dimensions or the configuration of a multi-rotor platform. The payload (onboard equipment) requirements of a single-UAV unit employing UVDAR are: an autopilot, a self-localization source (e.g. a GNSS receiver), 1–2 UV-sensitive cameras, computational power to control the flight and to process the data (one camera at 20 Hz requires approximately a 30% single-thread

load on Intel-Core i7 7567U, 3.5 GHz), and a set of UV LED markers placed at known extreme points of the UAV.

To verify this statement, an axiomatic functionality validation of UVDAR was performed on two independent multi-rotor platforms as shown in figure 5. The general hardware configuration of UAVs exhibited in the figure consists of

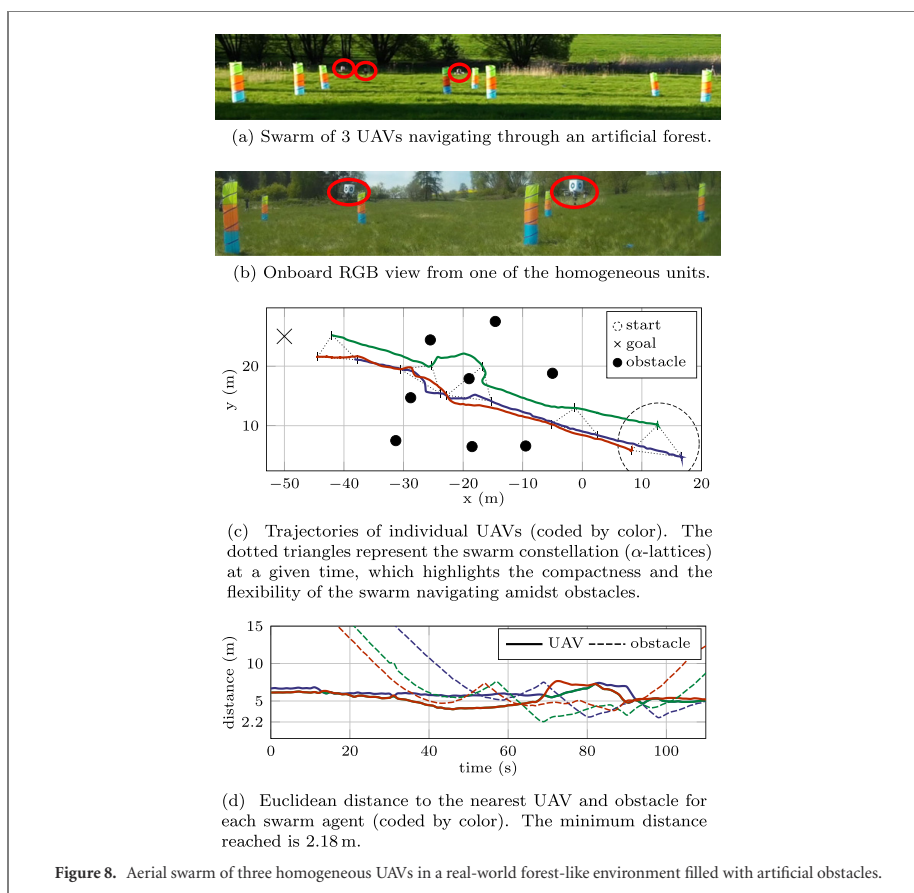


- the Pixhawk 4 autopilot,
- onboard computer Intel NUC i7 7567U,
- *ProLight Opto PM2B-1LLE* near-UV LEDs radiating at 390–410 nm wavelength [42],
- *mvBlueFOX-MLC* cameras with
 - * a *MidOpt BP365* near-UV band-pass filter and
 - * *Sunnex DSL215* fish-eye lenses,
 - a GNSS receiver (the hexa-rotor platform only), and
 - the Slamtec RPLiDAR-A3 laser scanner (the quad-rotor platform only).

The weight of this hardware configuration is 370 g (or 540 g with the laser scanner required either for an obstacle detection or for a local localization

replacing the GNSS dependency). The onboard Intel NUC computer weighing 225 g provides exaggerated processing power useful particularly in our case for general research purposes. For use in highly specialized applications, a feasible replacement of this payload with a microprocessor technology would allow for even further minimization of the aerial platform dimensions and cost expenses.

Further miniaturization of infrastructure/independent UAVs is limited by current technology required for local self-localization. Vision-based algorithms employ lightweight cameras minimizing the weight; however, it comes at the cost of high processing power and thus increased weight of the processing unit. On the other hand, laser-based localization generally requires less processing



power, but the sensors are heavier than cameras—approximately 170 g for planar scanners and 475 g for 3D LiDARs.

6. Experimental analysis

The primary aim of the experimental analysis is to verify the general functionality and to evaluate the performance of the entire framework exploiting direct localization rather than communication. The objectives of the experiments are focused primarily on determining the accuracy of the UVDAR direct localization, and on the stabilization and spatial navigation of an aerial swarm in real-world environments with and without obstacles. The entire experimental analysis is supported by multimedia materials available at <http://mrs.felk.cvut.cz/research/swarm-robotics>.

6.1. Swarming model analysis

To rule out the influence of UVDAR in a position control feedback loop of an aerial swarm, the *Boids*-based swarming intelligence (see section 4) is analyzed independently from the direct localization. For this purpose, the UAVs replace direct visual localization by sharing their global GNSS positions in an

ad-hoc network in order to determine the relative arrangement in the local neighborhood. This configuration was necessary in order to deploy UAVs without direct localization using UVDAR, as discussed in section 1.1. The analysis showcases the usability of the proposed fully-decentralized swarming framework both in simulations and in real-world scenarios, and in environments with and without obstacles. The global positions of the obstacles are *a priori* available to the UAVs.

First, the collective dynamics of the swarming model are analyzed thoroughly in the Gazebo robotic simulator [57], shown in figure 3, coupled with the robot operating system [58]. This simulation environment emulates real-world physics, and allows us to use identical low-level controllers and state estimation methods (see section 5) for the real UAVs and also for the simulated UAVs, without simplifying assumptions. This makes the configuration ideal for effortless deployment of theoretical bio-inspired swarming approaches onto a group of real-world robots. Simulation deployment of a swarm of homogeneous units in a 3D environment with obstacles (see

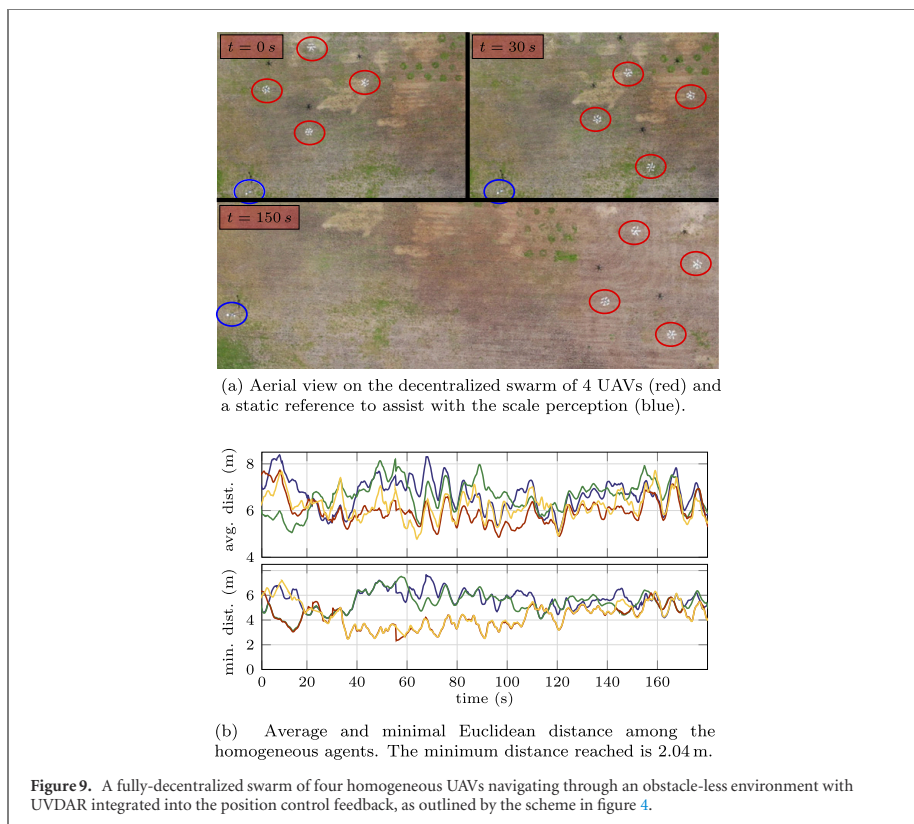


figure 6) verifies the qualitative performance of the reactive obstacle avoidance methodology presented in section 4. The emerging collective dynamics show the properties of the 3D shape flexibility during navigation through a narrow passage and in collision-free bypassing of static obstacles. The properties of safe navigation and high flexibility are also showcased during the simulation deployment of a compact swarm of nine homogeneous units in a dense 3D forest-like environment, according to figure 7.

Second, an aerial swarm of three UAVs was experimentally deployed in a real-world forest-like environment similar to figure 7, in order to verify the abilities of the fully-decentralized swarming model to stabilize a set of UAVs in a decentralized manner, provide self-organizing behavior, and to navigate through an obstacle-filled environment. As explicitly shown in figure 8, even such a simplistic swarming model with only local information yields collision-free navigation (the minimum distance to an obstacle or to another UAV was 2.2 m) throughout the environment, and self-organizing compactness of the whole swarm during the entire flight. The experiment likewise shows the ability of the model to divide the group when overcoming an obstacle and to unite back

again afterwards. This level of flexibility is important for fast and safe navigation within more complex environments in order to maximize the motion effectiveness. The flexibility is highlighted by dotted triangles, which represent the geometric configuration of the swarm in time. Let us call this flock geometry an α -lattice according to [45] and use it to represent a self-organizing structure, where individual inter-particle distances converge to a common value. This geometric configuration allows for small deviations from the expected structure (especially for particles in an environment with obstacles), which can be further quantified by *deviation energy* and can be used to evaluate the swarming model convergence. The deviation energy is derived in [45] and represents a non-smooth potential function of a set of particles, where the α -lattice configuration lies at its global minimum.

6.2. UVDAR in control feedback

To verify the feasibility of the complete system defined in figure 4, UVDAR vision-based mutual relative localization is deployed in the position control feedback loop of each homogeneous swarm agent. Throughout the experiment, the individual UAVs employ GNSS for self-state estimation. This is

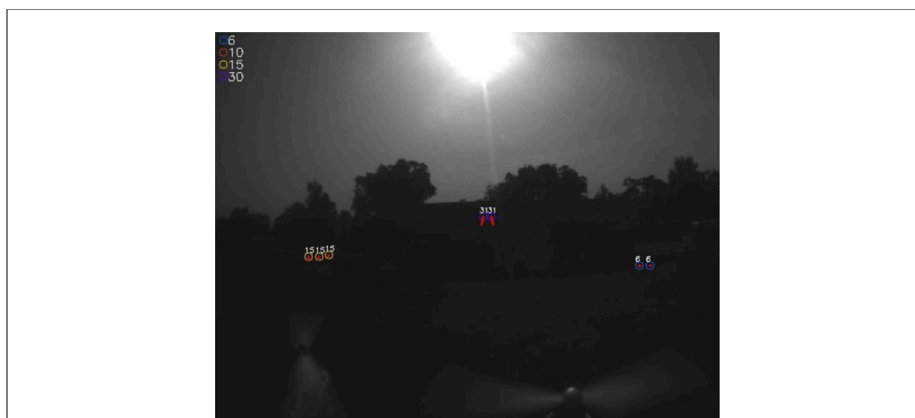


Figure 10. Onboard detection of three UAVs in the UV spectrum using UVDAR in a member of the aerial swarm. The method directly localizes the neighbors within a swarm in indoor and outdoor environments. Here, the method detects neighbors in an outdoor environment affected by a powerful source of ambient UV radiation. The processing is possible due to periodic blinking of the members with a specific frequency, here with 6 Hz, 15 Hz and 30 Hz.

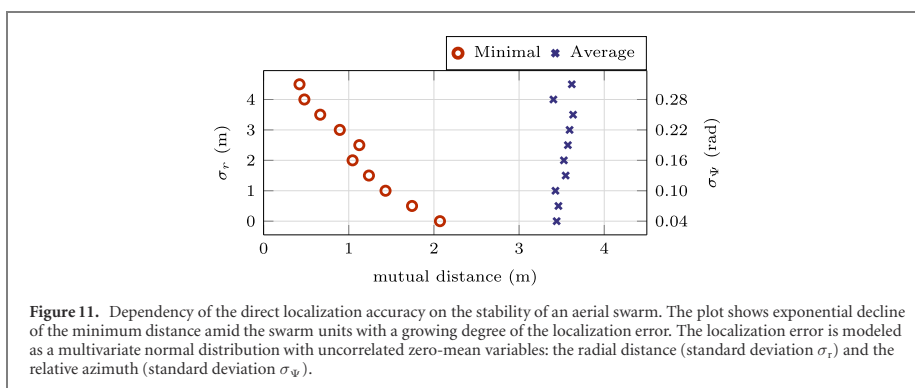


Figure 11. Dependency of the direct localization accuracy on the stability of an aerial swarm. The plot shows exponential decline of the minimum distance amid the swarm units with a growing degree of the localization error. The localization error is modeled as a multivariate normal distribution with uncorrelated zero-mean variables: the radial distance (standard deviation σ_r) and the relative azimuth (standard deviation σ_ψ).

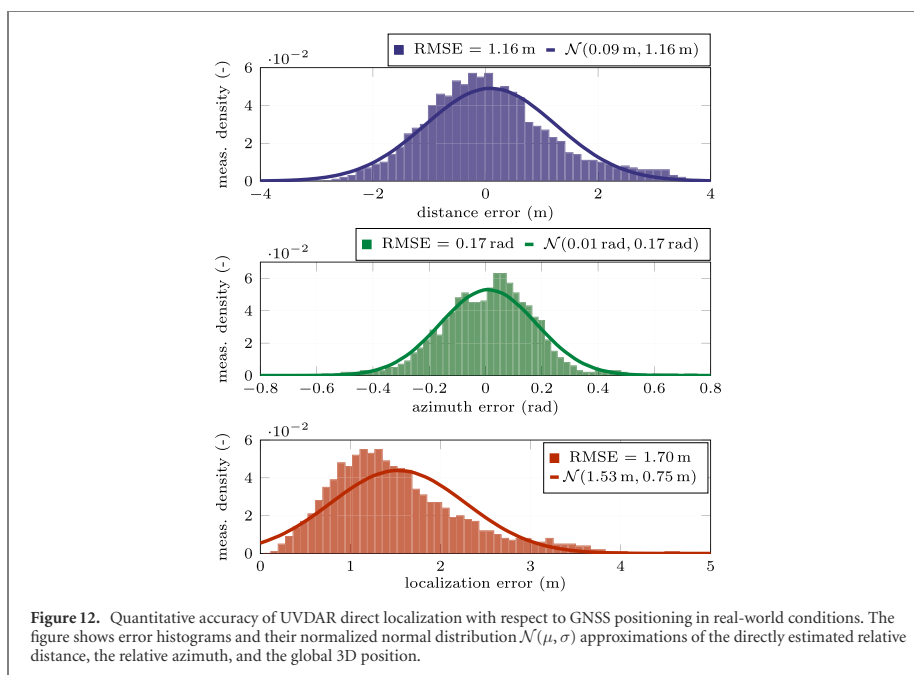
required to stabilize the flight of each dynamically unstable UAV mid-flight in a large open-space, where the swarm was deployed. However, the agents do not share any information through a communication network and instead they directly perceive the neighboring particles using UVDAR. The blinking frequencies of the UAVs (IDs) within the experiment were static and unique. This improves the performance of the UVDAR localization as unique IDs in the image stream help to separate occluded detections and track the units in time. To the best of our knowledge, this is the first deployment of a fully decentralized aerial swarming system in a real environment (outside laboratory-like conditions) with direct localization and with no communication or position sharing allowed.

As explicitly shown in figure 9, use of a local sensing method maintains the abilities of the bio-inspired swarming model, namely self-organizing behavior, together with collision-free and cohered navigation. The swarm is capable of navigation throughout the

environment in a compact structural constellation without any external interference to a sequence of global navigation goals. The figure shows the ability to preserve a compact structure emerging from local UVDAR-based perception (figures 2 and 10 show the perceived data of a single swarm agent in this particular experiment) and the elementary rules presented in section 4, while the homogeneous units do not share any information among themselves.

6.3. Analysis on direct observation accuracy

In real-world conditions, all estimation subsystems are incorporated with various measurements containing a stochastic noise element. The origin of this stochastic part is of numerous types (e.g. vibrations, discretization, approximations, sensor non-linearity, time desynchronization, lack of motion compensation, or optical discrepancies) and most of these inaccuracies need to be accounted for. For example, the stabilization and control system of UAVs requires a continuous stream of inertial measurements to



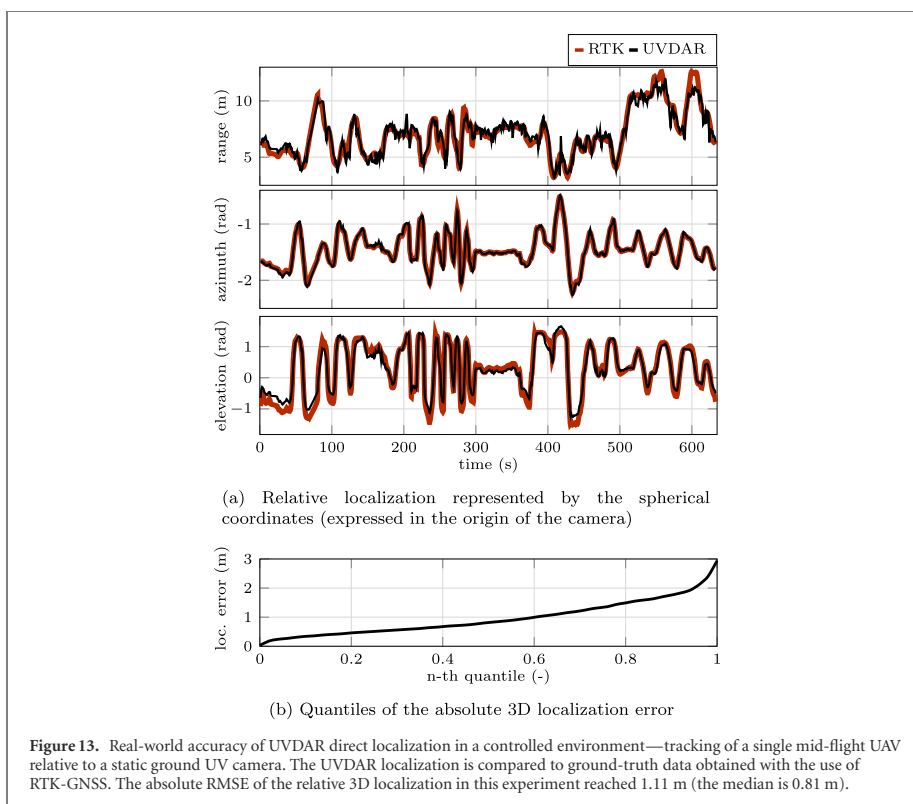
cope with hardware-based and synchronization inaccuracies, in order to stabilize the dynamically unstable system in mid-flight. The influence of these inaccuracies needs to be carefully analyzed, and the results of the analyses must be incorporated into the design of a swarming model in order to compensate for the uncertainties of real-world systems.

As discussed in the review of the related literature (see section 1.1), dense robotic swarms candidly communicate either external positioning estimates or individual global state estimates amid the swarm units. In addition to the requirements of the communication infrastructure, this methodology imitates the bio-inspired design of mutual localization by establishing the relative relations from the global data. This incorporates the global self-localization error, and can lead to dangerous decision making, and also to communication-based failures. However, our approach imitates biological systems by relying solely on direct localization without the need for known global states of the neighbors or of the unit itself. This bounds the overall performance of the system solely to the accuracy of the direct localization. It entirely removes the need for a communication infrastructure, and allows for full decentralization of the system architecture.

To analyze the impact of direct localization accuracy on the overall performance of our swarming framework, we present two inquiries: the influence of the error degree on the stability of a decentralized swarm, and the data-based accuracy of UVDAR in real-world conditions. As our focus applies to

vision-based direct localization, the error of 3D relative localization can be expressed in spherical coordinates—radial distance, azimuth, and elevation—separately. Bear in mind that due to the vision-based nature of UVDAR discussed in section 3, the statistical characteristics of the elevation error are assumed to be identical with the azimuthal error. To maintain simplicity, the elevation error is therefore omitted from the presentation of the results.

The impact of a direct localization error on the stability of a swarm was analyzed on a set of computational simulations. A decentralized swarm of UAVs with simulated dynamics, control & state estimation disturbances, and sensory inaccuracies, was deployed in scenarios with various degrees of the direct localization error according to figure 11. Although the data show the minimum influence of the error on the average distance among the swarm units, the stochastic element induces oscillations of the mutual distances. These deviations from a consensual mutual distance arise directly from the inaccuracy of direct localization and from time-based and dynamics-based delays. This has a negative impact on the stability properties of the entire swarm, as shown by the exponential decline of the minimal distance amid the swarm units with the increasing degree of the radial distance and the relative azimuth error in figure 11. In real-world systems, a suitable swarm density must be thoroughly considered with respect to the accuracy and the reliability of the direct localization in order to prevent undesired collisions.



The accuracy of UVDAR in real-world conditions during the deployment of the decentralized swarm of four UAV units in an open environment (see figure 9) is expressed by the error histograms in figure 12. During this experiment, the self-localization of the individual UAVs was arranged by GNSS. The statistical analysis uses global positioning for a quantitative evaluation of the direct localization accuracy. Although global positioning yields a relatively high error, the state estimation module (see section 5) fuses this global state estimate with inertial measurements, which makes the output estimate robust towards sudden short-term changes. The positioning is still prone to long-term drift, which is minimal in terms of GNSS and therefore does not significantly impact the evaluation of the direct localization within a dense swarm. The fused global estimate is therefore used as ground truth data for the quantitative evaluation in figure 12. This evaluation on real-world data shows the ability of UVDAR to estimate the relative distance with 1.16 m root mean square error (RMSE) and the relative azimuth with RMSE of 0.17 rad. These separated errors then combine together with the elevation estimate to anticipate the relative 3D position of the neighboring particles within a moving aerial swarm with RMSE of 1.7 m.

The accuracy of UVDAR in real-world conditions is further analyzed in a controlled outdoor environment. During an independent experiment, a position of a single mid-air UAV was tracked in data from a static ground camera equipped with UVDAR and was compared to a precise RTK-GNSS (2 cm accuracy) serving as a ground-truth. The comparison of the relative localization with the ground-truth data is shown in figure 13. The data show the property of UVDAR to localize an aerial unit with RMSE of 1.11 m.

The concluded accuracy is particularly important for the design of bio-inspired systems employing the UVDAR sensor as a source of direct localization of neighboring units. The quantitative results of this analysis allow for appropriate compensation of the inaccuracies and credible verification of swarming models in a simulator, which necessarily precede real-world applications.

7. Conclusion

This article has presented a framework for deploying fully-decentralized aerial swarms in real-world conditions with the use of vision-based UV mutual relative localization of neighboring swarm units. The framework architecture, as well as the off-the-shelf UVDAR

system for direct localization within an aerial swarm, has been thoroughly discussed, has been deployed on a decentralized swarm of UAVs in real-world environments, and its performance has been analyzed. The experimental analysis verified the stability of UVDAR as an input into a fully-decentralized swarming architecture, which embodies the communication-free and local-information swarming models that are commonly found among biological systems. The set of real-world experiments is, to the best of our knowledge, the first deployment of a decentralized swarm of UAVs with no use of a communication network or of external localization. The system is provided as open source, and is designed for simple integration and verification of flocking techniques (often bio-inspired), respecting the requirements of the swarming paradigm.

Acknowledgments

This work was supported by the Czech Science Foundation (GAČR) under research project No. 20-10280S, by CTU Grant No. SGS20/174/OHK3/3T/13, by funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 871479, and by OP VVV project CZ.02.1.01/0.0/0.0/16 019/0000765 'Research Center for Informatics'. The authors thank Daniel Heřt for preparing all the necessary equipment required for the experimental analysis.

ORCID iDs

Pavel Petráček  <https://orcid.org/0000-0002-0887-9430>

Viktor Walter  <https://orcid.org/0000-0001-8693-6261>

Tomáš Báča  <https://orcid.org/0000-0001-9649-8277>

Martin Saska  <https://orcid.org/0000-0001-7106-3816>

References

- [1] Štibinger P, Baca T and Saska M 2020 Localization of ionizing radiation sources by cooperating micro aerial vehicles with pixel detectors in real-time *IEEE Robot. Autom. Lett.* **5** 3634–41
- [2] Gassner M, Cieslewski T and Scaramuzza D 2017 Dynamic collaboration without communication: vision-based cable-suspended load transport with two quadrotors *Int. Conf. on Robotics and Automation* pp 5196–202
- [3] Trianni V 2008 *Evolutionary Swarm Robotics: Evolving Self-Organising Behaviours in Groups of Autonomous Robots (Studies in Computational Intelligence)* 1st edn (Berlin: Springer)
- [4] Young G F, Scardovi L, Cavagna A, Giardina I and Leonard N E 2013 Starling flock networks manage uncertainty in consensus at low cost *PLOS Comput. Biol.* **9** 1–7
- [5] Chung S-J, Paranjape A A, Dames P, Shen S and Kumar V 2018 A survey on aerial swarm robotics *IEEE Trans. Robot.* **34** 837–55
- [6] Nguyen T-M, Qiu Z, Nguyen T H, Cao M and Xie L 2019 Distance-based cooperative relative localization for leader-following control of MAVs *IEEE Robot. Autom. Lett.* **4** 3641–8
- [7] Bhavana T, Nithya M and Rajesh M 2017 Leader-follower co-ordination of multiple robots with obstacle avoidance *SmartTechCon*
- [8] Saska M, Baca T, Thomas J, Chudoba J, Preucil L, Krajník T, Faigl J, Loianno G and Kumar V 2017 System for deployment of groups of unmanned micro aerial vehicles in GPS-denied environments using onboard visual relative localization *Auton. Robot.* **41** 919–44
- [9] Faigl J, Krajník T, Chudoba J, Přeucil L and Saska M 2013 Low-cost embedded system for relative localization in robotic swarms *Int. Conf. on Robotics and Automation* pp 993–8
- [10] De Silva O, Mann G K I and Gosine R G 2015 An ultrasonic and vision-based relative positioning sensor for multirobot localization *IEEE Sensors J.* **15** 1716–26
- [11] Yan X, Deng H and Quan Q 2019 Active infrared coded target design and pose estimation for multiple objects *Int. Conf. on Intelligent Robots and Systems* pp 6885–90
- [12] Censi A, Strubel J, Brandli C, Delbruck T and Scaramuzza D 2013 Low-latency localization by active LED markers tracking using a dynamic vision sensor *Int. Conf. on Intelligent Robots and Systems* pp 891–8
- [13] Park H, Choi I, Park S and Choi J 2013 Leader-follower formation control using infrared camera with reflective tag *10th Int. Conf. on Ubiquitous Robots and Ambient Intelligence* pp 321–4
- [14] Chaudhary K, Zhao M, Shi F, Chen X, Okada K and Inaba M 2017 Robust real-time visual tracking using dual-frame deep comparison network integrated with correlation filters *Int. Conf. on Intelligent Robots and Systems* pp 6837–42
- [15] Carrio A, Tordesillas J, Vemprala S, Saripalli S, Campoy P and How J P 2020 Onboard detection and localization of drones using depth maps *IEEE Access* **8** 30480–90
- [16] Intel® 2019 Drones light up the sky URL <http://intel.com/content/www/us/en/technology-innovation/aerial-technology-light-show.html>
- [17] Virágh C, Vásárhelyi G, Tarcai N, Szörényi T, Somorjai G, Nepusz T and Vicsek T 2014 Flocking algorithm for autonomous flying robots *Biomim. Biomim.* **9** 025012
- [18] Vásárhelyi G, Virágh C, Somorjai G, Tarcai N, Szörényi T, Nepusz T and Vicsek T 2014 Outdoor flocking and formation flight with autonomous aerial robots *Int. Conf. on Intelligent Robots and Systems* pp 3866–73
- [19] Vásárhelyi G, Virágh C, Somorjai G, Nepusz T, Eiben A E and Vicsek T 2018 Optimized flocking of autonomous drones in confined environments *Sci. Robot.* **3** eaat3536
- [20] EHang 2019 EHang drone formation flight URL <http://ehang.com/formation>
- [21] Hauert S, Leven S, Varga M, Ruini F, Cangelosi A, Zufferey J and Floreano D 2011 Reynolds flocking in reality with fixed-wing robots: communication range vs maximum turning rate *Int. Conf. on Intelligent Robots and Systems* pp 5015–20
- [22] Bürkle A, Segor F and Kollmann M 2011 Towards autonomous micro UAV swarms *J. Intell. Robot. Syst.* **61** 339–53
- [23] Kushleyev A, Mellinger D, Powers C and Kumar V 2013 Towards a swarm of agile micro quadrotors *Auton. Robot.* **35** 287–300
- [24] Weinstein A, Cho A, Loianno G and Kumar V 2018 Visual inertial odometry swarm: an autonomous swarm of vision-based quadrotors *IEEE Robot. Autom. Lett.* **3** 1801–7
- [25] Stirling T, Roberts J, Zufferey J C and Floreano D 2012 Indoor navigation with a swarm of flying robots *Int. Conf. on Robotics and Automation* pp 4641–7

- [26] Nägele T, Conte C, Domahidi A, Morari M and Hilliges O 2014 Environment-independent formation flight for micro aerial vehicles *Int. Conf. on Intelligent Robots and Systems* pp 1141–6
- [27] van Diggelen F and Enge P K 2015 The world's first GPS MOOC and worldwide laboratory using smartphones *ION GNSS+* pp 361–9
- [28] Garcia Carrillo L R, Fantoni I, Rondon E and Dzul A 2015 Three-dimensional position and velocity regulation of a quad-rotorcraft using optical flow *IEEE Trans. Aerosp. Electron. Syst.* **51** 358–71
- [29] Schmid K, Lutz P, Tomić T, Mair E and Hirschmüller H 2014 Autonomous vision-based micro air vehicle for indoor and outdoor navigation *J. Field Robot.* **31** 537–70
- [30] Kohlbrecher S, Meyer J, von Stryk O and Klingauf U 2011 A flexible and scalable SLAM system with full 3D motion estimation *Int. Symp. on Safety, Security and Rescue Robotics* pp 155–60
- [31] Schmickl T and Hamann H 2016 *BEECLUST: A Swarm Algorithm Derived from Honeybees: Derivation of the Algorithm, Analysis by Mathematical Models, and Implementation on a Robot Swarm* pp 95–137 (Boca Raton, FL: CRC Press)
- [32] Bodi M, Möslinger C, Thenius R and Schmickl T 2015 BEECLUST used for exploration tasks in autonomous underwater vehicles *8th Int. Conf. on Mathematical Modelling* pp 819–24
- [33] Shah D and Vachhani L 2019 Swarm aggregation without communication and global positioning *IEEE Robot. Autom. Lett.* **4** 886–93
- [34] Olaronke I, Rhoda I, Gambo I, Oluwaseun O and Janet O 2020 A systematic review of swarm robots *Curr. J. Appl. Sci. Technol.* **39** 79–97
- [35] Oh H, Shirazi A R, Sun C and Jin Y 2017 Bio-inspired self-organising multi-robot pattern formation: a review *Robot. Autom. Syst.* **91** 83–100
- [36] Smith N M, Dickerson A K and Murphy D 2019 Organismal aggregations exhibit fluidic behaviors: a review *Bioinspir. Biomim.* **14** 031001
- [37] Saska M, Vakula J and Přeucil L 2014 Swarms of micro aerial vehicles stabilized under a visual relative localization *Int. Conf. on Robotics and Automation* pp 3570–5
- [38] Walter V, Staub N, Saska M and Franchi A 2018 Mutual localization of UAVs based on blinking ultraviolet markers and 3D time-position Hough transform *14th Int. Conf. on Automation Science and Engineering* pp 298–303
- [39] Walter V, Staub N, Franchi A and Saska M 2019 UVDAR system for visual relative localization with application to leader-follower formations of multirotor UAVs *IEEE Robot. Autom. Lett.* **4** 2637–44
- [40] Walter V, Vrba M and Saska M 2020 On training datasets for machine learning-based visual relative localization of micro-scale UAVs *Int. Conf. on Robotics and Automation* accepted
- [41] International Commission on Non-Ionizing Radiation Protection and others 2004 Guidelines on limits of exposure to ultraviolet radiation of wavelengths between 180 nm and 400 nm (incoherent optical radiation) *Health Phys.* **87** 171–86
- [42] Stanford Artificial Intelligence Laboratory et al 2018 Robotic Operating System <https://www.ros.org>
- [43] Calovi D S, Lopez U, Ngo S, Sire C, Chaté H and Theraulaz G 2014 Swarming, schooling, milling: phase diagram of a data-driven fish school model *New J. Phys.* **16** 015026
- [44] Reynolds C W 1987 Flocks, herds and schools: a distributed behavioral model *14th Annual Conf. on Computer Graphics and Interactive Techniques* pp 25–34
- [45] Olfati-Saber R 2006 Flocking for multi-agent dynamic systems: algorithms and theory *IEEE Trans. Autom. Control* **51** 401–20
- [46] Zhu H, Juhl J, Ferranti L and Alonso-Mora J 2019 Distributed multi-robot formation splitting and merging in dynamic environments *Int. Conf. on Robotics and Automation* pp 9080–6
- [47] Erunsal I K, Ventura R and Martinoli A 2019 Nonlinear model predictive control for 3d formation of multirotor micro aerial vehicles with relative sensing in local coordinates (arXiv:1904.03742)
- [48] Curiaç D-I and Volosencu C 2015 Imparting protean behavior to mobile robots accomplishing patrolling tasks in the presence of adversaries *Bioinspir. Biomim.* **10** 056017
- [49] Elamvazhuthi K and Berman S 2019 Mean-field models in swarm robotics: a survey *Bioinspir. Biomim.* **15** 015001
- [50] Alonso-Mora J 2014 Collaborative motion planning for multi-agent systems *PhD Thesis* Autonomous Systems Lab, ETH-Zürich
- [51] Mohamed E F, El-Metwally K and Hanafy A R 2011 An improved tangent bug method integrated with artificial potential field for multi-robot path planning *Int. Symp. on Innovations in Intelligent Systems and Applications* pp 555–9
- [52] Báča T, Petrлік M, Vrba M, Spurný V, Pěnicka R, Hert D and Saska M 2020 The MRS UAV system: pushing the frontiers of reproducible research, real-world deployment, and education with autonomous unmanned aerial vehicles (arXiv:2008.08050)
- [53] Lee T, Leok M and McClamroch N H 2010 Geometric tracking control of a quadrotor UAV on SE(3) *49th Conf. on Decision and Control* pp 5420–5
- [54] Petráček P, Krátký V and Saska M 2020 Dronument: system for reliable deployment of micro aerial vehicles in dark areas of large historical monuments *IEEE Robot. Autom. Lett.* **5** 2078–85
- [55] Báča T, Hert D, Loianno G, Saska M and Kumar V 2018 Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles *Int. Conf. on Intelligent Robots and Systems* pp 6753–60
- [56] Meier L, Tanskanen P, Heng L, Lee G H, Fraundorfer F and Pollefeys M 2012 PIXHAWK: a micro aerial vehicle design for autonomous flight using onboard computer vision *Auton. Robot.* **33** 21–39
- [57] Koenig N and Howard A 2004 Design and use paradigms for Gazebo, an open-source multi-robot simulator *Int. Conf. on Intelligent Robots and Systems* vol 3 pp 2149–54
- [58] Stanford Artificial Intelligence Laboratory et al 2018 Robotic Operating System available at: <https://www.ros.org>

■ 5 Discussion and Summary of Results

In this chapter, we summarize and discuss the author’s overall contributions with respect to the research challenges and objectives defined in Chapter 1.

■ 5.1 LiDAR-based Perception and UAV Autonomy in Demanding Real-World Conditions

Challenge (1) and objective (1) have been addressed in five out of six of the core publications and in a plethora of other co-authored works. In parallel, all of these publications also contribute to challenge (3) and objective (3). These challenges and objectives emphasize that state-of-the-art aerial robotics is hindered by unsatisfactory perceptual robustness in challenging real-world conditions devoid of GNSS. The two main challenges related to the objectives (1) and (3) have been defined and focused on throughout the thesis — operating under perceptual degeneracy and developing methods for resource-constrained robots. In this regard, the proposed contributions summarized below go beyond the state-of-the-art LiDAR-based perception by introducing novel methods for achieving satisfactory robustness and perceptual resiliency of aerial systems.

Given the projective nature of modern 3D LiDARs, the perceptual measurement can be structured such that the estimation of the ego-motion of the robot is challenging, if even possible. This perceptual degeneracy is connoted as perceptual (or geometrical) aliasing and emerges in geometrically symmetrical and structureless environments. In these environments, the degeneracy is the main source of estimation drift. As demonstrated in Figure 2.1, such degeneracy can then be classified as weak and strong. In the core publication [1c], we have **tackled the harder problem of strong degeneracy in environments that are geometrically symmetrical** along the vertical axis. In particular, we focused on the interiors of large historical monuments, such as cathedrals and churches. To fight the vertical degeneracy, we introduced additional constraints to increase perceptual observability along the degenerate axis. Although grounded on cheap and lightweight 1D and 2D LiDARs, the enhanced observability of the proposed multi-sensor scheme allowed for fast and reliable 6-DoF pose estimation in the 3D world. The low amount of data from the lightweight sensors allowed for pose estimation on-board small-factor UAVs, leading to the first-ever autonomous mission of UAVs performing a documentation technique RTI [9a] in areas high above ground in a historic structure.

The challenging task of [1c] has been extended to multi-UAV configurations in core [2c] and co-authored [10a] publications. Building on experience gained in [1c, 9a, 131], these works rework the perceptual capabilities of [1c] into multi-sensor design, fusing a 3D LiDAR, LiDAR rangefinder, and inertial measurements. The primary source of pose estimation relies on a 6-DoF odometry utilizing the 3D LiDAR data. Although not fully coping with the challenge of strong degeneracy, the proposed methodology utilizes a map of the environment to detect drift in the estimation. This approach is not completely robust to the drift but introduces **awareness to geometrical degeneracy**, allowing for on-the-go alteration of the aerial mission to maximize safety. The methodology is designed for decentralized multi-UAV teams co-operating within a known environment full of obstacles. The work [2c] is not limited to historical settings, although it has been investigated in them due to the need for multi-robot solutions in the field of cultural preservation. As part of [2c], we have investigated and listed documentation techniques practicable by UAV teams and advanced the selected methods to

large-scale settings outside of the laboratory. The outcomes of [2c, 9a] include achievements in full real-world autonomy, as well as the first ever fully-autonomous inspection and documentation missions of a multi-UAV team in historical monuments, while performing tasks infeasible for a single UAV in principle (e.g., RTI [9a] and three-point lighting). Selected documentation missions from five different historical structures are demonstrated in Figure 5.1.

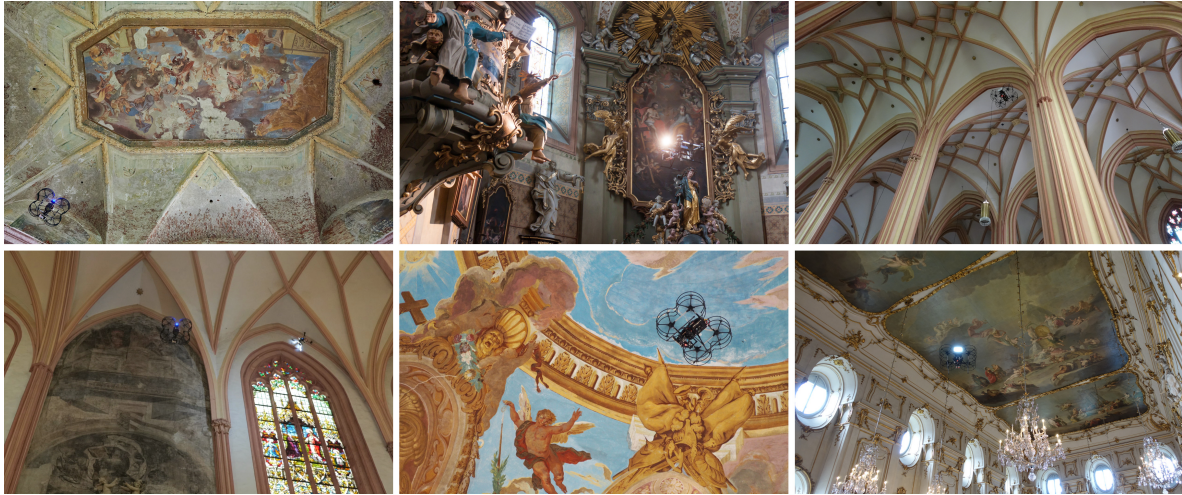


Figure 5.1: Selected illustrations of the theoretical contributions of [1c, 2a, 9a, 10a] as actualized to the real-world robotic autonomy, Dronument [137]. During the development and investigation phases, the research was applied for autonomous documentation of the interiors of 18 historical monuments, including two sites on the UNESCO World Heritage list. The work is supported by multimedia available at youtu.be/Gx-mBklSbYc.

Motivated by a lack of degeneracy-aware perception and the need for minimizing computational delay in pose estimation pipelines running on-board robots with limited computational resources, the core publication [6c] proposed a **novel method for sampling points measured by rotating 3D LiDAR sensors**. Rooted in theory stating that hyperplane surfaces generate residuals redundant in a pose estimation task, the method [6c] **quantifies and minimizes redundancy in the point cloud** by removing uninformative points together with noise and outliers. Since sampling at the input space of an estimation pipeline cannot relate to information theory, the method is unable to directly quantify the degree of degeneracy. Nevertheless, it outperforms the baseline state-of-the-art methods in well-conditioned settings as well as under weak degeneracy. It samples the least amount of points, yields the highest computational gains, and preserves the highest ratio of information extractable from the point cloud. Although [6c] is of principal importance to resource-constrained robots, this fundamental advance has the potential to improve all existing real-time estimation pipelines employing rotating 3D LiDARs, whether it is applied on-board an autonomous car or a UAV. This core work is supported by multimedia available at github.com/ctu-mrs/RMS.

On-board aerial perception and autonomy are challenging, even when the type and parameters of the environment are known beforehand (e.g., in [2c]). As motivated by the DARPA SubT (one of the most important competitions in mobile robotics), a main challenge in our other core and co-authored publications [3c, 4a, 8a] has been to push the state of the art **beyond known environments to unpredictable, harsh, and unknown settings**. Starting in [3c, 8a], we have advanced the fundamental theory of [39] into the real 3D world, providing a UAV the ability to perceive, plan, and act in the real world only on the basis of its own onboard sensors. The advancement removed any severe restrictions and 2D world

assumptions; thus, allowing to fully exploit the potential of agile multi-rotor UAVs. The perceptual capabilities of UAVs in these works build on the contributions related to the methodology of inspecting historical interiors [2c]. Nevertheless, the methodology of [3c, 4a, 8a] was exceedingly challenged by the variety of unpredictable difficulties of a dynamic 3D environment, as illustrated in Figure 5.2. The contributions of [3c] include a **full-stack, field-tested, and open-source LI-based UAV odometry and optimal multi-UAV homing** capable of autonomous operation in an arbitrary exploratory task within an unknown and unstructured subterranean environment. The experimental investigations in [3c] achieved deep-exploratory flights up to a length of 470 m in a single fully autonomous deployment in the Bull Rock Cave system, which is an extreme performance, even by today’s standards in field robotics. The work describing our research endeavor in S&R [4c] summarizes our contributions, including novel methods for robust dust and mist filtering from LiDAR data, thin-obstacle avoidance, tight coupling of all autonomy modules running on-board a UAVs (an important, yet often underestimated quality), and a multi-robot navigation stack capable of optimal multi-UAV exploration of unknown and dynamic environments. Deploying our top-performing UAV stack [3c, 4a, 8a] in DARPA SubT served as the ultimate test on the author’s contributions to the field related to the resiliency of UAV perception and autonomy.



Figure 5.2: Selected illustrations of the theoretical contributions on UAV autonomy of [3c, 4a, 8a, 39] as actualized in the real-world robotic system deployed in the DARPA SubT by team CTU-CRAS-NORLAB. A wide range of unpredictable real-world environments (narrow corridors, large caverns, tight openings, vertical shafts, dark, dusty and misty conditions, and many others) requires a high level of adaptability, resiliency, and robustness in on-board perception and autonomy. The work is summarized in multimedia available at youtu.be/8rLxX3kS9Qo.

■ 5.2 Distributed Perception-Aware Multi-UAV Coordination

The challenge (2) and objective (2) on distributed coordination of multi-UAV teams are laid in one core and four co-authored works. These works emphasize research on coordinating real-world decentralized teams in which the robots decide their actions based on local perceptual information. The objective (2) targets increasing the robustness of UAV teams to single points of failure, such as centralization and communication. The core work [5c] proposed a **biologically-inspired self-organizing swarming model** respecting the swarming principle defined in [36]. Our investigations show that the model is capable of navigating a team of non-communicating UAVs in an outdoor environment filled with obstacles on just the basis of local vision-based and LiDAR-based perception. The work [5c] served as a baseline method for further research, improving upon this work by researching novel models introducing:

- local path planning in GNSS-restricted environments (such as forests) [15a, 117],
- evasion to dynamic and hostile objects [11a],
- splitting and cohesive merging maneuverability [14a],
- proximal-control theory [16a] previously studied in relation to non-holonomic differential drive robots, and
- path persistence and similarity features for improving collective navigation [118].

The works, including the core publication [5c], build upon the UVDAR [37, 38] localization system (developed at the MRS laboratory at FEE, CTU), which provides perceptual vision-based feedback in the form of relative position and bearing of friendly and visible neighboring UAVs. At the time of publishing, [5c] presented the first real-world UAV swarm capable of cohesive navigation among obstacles without communication. Although further research on this topic is required, these contributions have the potential to provide reliability in tasks requiring multi-UAV teams, such as the safety-critical documentation of historical monuments [2c].

■ 5.3 Highlights, Interdisciplinary Results, and Secondary Activities

The author's contributions are heavily motivated by existing challenges of real-world applicability. Since some of these challenges require interdisciplinary methodology, the contributions reach beyond the field of robotics. The first interdisciplinary overreach was made into the field of **speleology** in [3c], developed together with the Institute of Geodesy at Brno University of Technology. In [3c], we defined the potential use of UAVs for the mapping, exploration, and disaster response in natural cave systems. The developed system was used to map the Bull Rock Cave system located in the central Moravian Karst of the Czech Republic. The second interdisciplinary contribution of the author lies in the field of **cultural preservation**. The mutual work on the Dronument project [2c] with the Czech National Heritage Institute led to several achievements beyond standard doctoral work. In particular, the high societal impact of this project includes the following.

- A detailed methodology for the safe use and deployment of autonomous UAVs in the field of cultural preservation was specified as part of [2c]. The Czech Ministry of Culture authorized and published this methodology as an official hardware and software solutions rulebook in the related fields.
- The functional sample (see Figure 5.3) of this work was awarded in the M17+ 2022 evaluation [130] for excellent scientific achievements in the Czech Republic.
- The work [2c] describes and lists documentation tasks realizable and inapplicable by aerial vehicles. The list serves as a future reference for interdisciplinary work and includes

techniques, such as multi-spectral imaging, 3D reconstruction, photogrammetry, three-point lighting, raking light, reflectance transformation imaging, and many others.

- Part of the work was published in the Czech journal *Památky*, sharing the findings among experts in the field of cultural preservation in [18a].
- The developed technology has been used to document 18 historical monuments around the Czech Republic. These structures include two objects contained on the UNESCO World Heritage list. To this day, this technology remains the most comprehensive project in the field of autonomous documentation of historical monuments by an aerial system.

The work is also supported by results and multimedia materials available at [137]. These results contain the methodology, functional sample, and 3D models of selected structures documented by the developed technology. Note that the developed technology is not limited to historical settings and is easily adaptable to other scenarios, especially in industrial settings.



Figure 5.3: Functional sample of project Dronument awarded in the M17+ 2022 evaluation [130] for excellent scientific achievements in the Czech Republic.

The author participated in the renowned DARPA SubT as a member of team CTU-CRAS-NORLAB, which won the following.

- First place (third overall) among non-funded teams in the Tunnel Circuit (US\$200k)¹.
- Second place (third overall) among non-funded teams in the Urban Circuit (US\$500k).
- Second place among all teams in the Virtual Track of the Final Circuit (US\$500k).

The author's work on LiDAR-based perception and multi-robot S&R tasks contributed to the receipt of 2 million US\$ of funding for the Center for Robotics and Autonomous Systems (CRAS) from the US DARPA for further research on the field-robotics methodology applicable to S&R tasks.



Figure 5.4: Team CTU-CRAS-NORLAB at the Urban Circuit (left) and the Final Circuit (right) of the DARPA Subterranean Challenge.

¹Author of this thesis did not participate in this round.

Additionally, the co-authored high-impact work [7a] builds upon the experience gained by top teams during the DARPA SubT and summarizes the current state of the art and ongoing challenges in field robotics regarding perception and SLAM in extreme environments. Together with other works [6c, 14a], the work [7a] shows the capabilities of the author of this thesis to cooperate internationally with the leading experts in the field of mobile robotics.

The author’s secondary academic activities include the following workshop presentations at IEEE International Conference on Intelligent Robots and Systems 2018 and 2023.

- P. Petracek, V. Kratky, M. Petrlik, and M. Saska, “Cooperative UAV Autonomy of Dronument: New Era in Cultural Heritage Preservation,” in *IEEE IROS (Workshop on Integrated Perception, Planning, and Control for Physically and Contextually-Aware Robot Autonomy)*, 2023.
- P. Petracek, N. Khedekar, M. Nissov, K. Alexis, and M. Saska, “Importance Sampling: Degradation-Aware Alternative to Voxelization in Robot Pose Estimation,” in *IEEE IROS (Workshop on Robotic Perception and Mapping: Frontier Vision & Learning Techniques and Workshop on Integrated Perception, Planning, and Control for Physically and Contextually-Aware Robot Autonomy)*, 2023.
- P. Petráček and M. Saska, “Decentralized Aerial Swarms Using Vision-Based Mutual Localization,” in *IEEE IROS (Second Workshop on Multi-robot Perception-Driven Control and Planning)*, 2018.

The author’s work during his doctoral studies also supported the community through academic endeavors, contributing to the open-source educational MRS UAV System [39] and co-organizing an international 2021 Dronument workshop attended by 80 people, as well as the 2022 IEEE Robotics and Automation Society (RAS) Summer School on Multi-Robot Systems, which was attended by 180 international students [19a] (see Figure 5.5).



Figure 5.5: Participants of the 2022 IEEE RAS Summer School on Multi-Robot Systems [19a], which was motivated by the results achieved in [2c] and [4c].

Lastly, the authored contributions have factored into considerable public interest, notably popularizing scientific endeavors in mobile robotics at CTU among general audiences. A standout project, Dronument, has attracted significant media attention (full list available in [137]), with coverage including 11 television reports, some of which aired during prime time on Czech national television, along with over 35 newspaper articles showcasing our research methodologies and outcomes. Additionally, the involvement of CTU in the DARPA SubT has led to coverage in at least 2 television reports and over 15 newspaper articles in Czech media outlets, further enhancing the public’s awareness to the field.

■ 6 Conclusion

This thesis has addressed the topic of degraded perception as the primary bottleneck in reliable autonomy of small-factor UAVs. The impediment factors have been linked to perceptual degeneracy arising in environments that are geometrically symmetrical and structureless. To challenge this bottleneck, the contributions of this thesis include novel LiDAR-based solutions, including localization that maximizes observability in structures requiring critical safety, redundancy-minimizing sampling of 3D point clouds for drift-reducing robot-pose estimation in real-time, and robust filtering of whirling dust clouds emerging from the aerodynamic influence of a multi-rotor UAV in extreme settings. Additionally, centralization and communication were identified as points of failure in compact multi-UAV teams. Novel perception-aware swarming models were proposed to substitute these points of failure in order to maximize safety in tasks requiring multiple closely-cooperating robots. All the contributions yield minimal computational latency and are capable of operating onboard robots with constrained computational resources, as supported by a variety of demonstrations in unforgiving real-world settings. Furthermore, we highlight the actualization of theoretical contributions in UAV autonomy that was field-tested inside extreme subterranean environments during the search and rescue tasks of DARPA SubT, in natural cave systems as driven by the needs of speleology experts, and in interiors of large historical monuments as motivated by the interdisciplinary project Dronument. The proposed contributions served as an integral part in both the top-performing UAV-based autonomy stack in the SubT and the Dronument project, the latter of which has since been used to document 18 historical monuments, including two structures on the UNESCO World Heritage list. Furthermore, the contributions are part of high-impact publication [7a] defining the current and future directions of perception in mobile robotics. The thesis builds upon six core and thirteen co-authored publications that have contributed to the specified objectives and challenges of the current state of the art in mobile aerial robotics.

■ A Author's List of Publications

All of the author's peer-reviewed publications are listed below. The core publications of the thesis are referenced with [*c] and other author's publications with [*a]. The publications are listed together with the following information.

- Journal Impact Factor (JIF) at the time of publishing a journal article as by the Journal Citation Reports™, Clarivate. Categorization in Science Citation Index Expanded (SCIE) is listed unless specified otherwise.
- Percentage of the author's contributions as agreed among authors upon publication.
- Credit author statement defined as Contributor Roles Taxonomy (CRediT) by Elsevier¹. Only the CRediT statements of the author of this thesis are included.
- Number of citations (including self-citations) based on Web of Science (WoS) (if indexed), Scopus (if indexed), and Google Scholar (GS) gathered on March 27, 2024. A summary and full list of citations (excluding self-citations) is given in Appendix B.

■ A.1 Core peer-reviewed publications

■ A.1.1 Core journal articles with Journal Impact Factor

- [1c] **P. Petracek**, V. Kratky, and M. Saska, "Dronument: System for Reliable Deployment of Micro Aerial Vehicles in Dark Areas of Large Historical Monuments," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2078–2085, 2020.

JIF	3.7 (Q2 in Robotics)
Contributions	PP: 70% , VK: 20%, MS: 10%
CRediT	conceptualization, methodology, software, validation, formal analysis, investigation, writing, visualization
Citations	22 in WoS, 22 in Scopus, 46 in GS

- [2c] **P. Petracek**, V. Kratky, T. Baca, M. Petrlik, and M. Saska, "New Era in Cultural Heritage Preservation: Cooperative Aerial Autonomy for Fast Digitalization of Difficult-to-Access Interiors of Historical Monuments," *IEEE Robotics and Automation Magazine*, pp. 2–19, 2023. **Equal contribution of the first two co-authors.**

JIF	5.7 (Q1 in Automation and Control Systems)
Contributions	PP: 41% , VK: 41%, TB: 6%, MP: 6%, MS: 6%
CRediT	conceptualization, methodology, software, validation, formal analysis, investigation, data curation, writing, visualization, project administration
Citations	1 in WoS, 2 in Scopus, 9 in GS

- [3c] **P. Petracek**, V. Kratky, M. Petrlik, T. Baca, R. Kratochvil, and M. Saska, "Large-Scale Exploration of Cave Environments by Unmanned Aerial Vehicles," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7596–7603, 2021.

JIF	4.3 (Q2 in Robotics)
Contributions	PP: 42% , VK: 20%, MP: 20%, TB: 6%, RK: 6%, MS: 6%
CRediT	conceptualization, methodology, software, validation, formal analysis, investigation, writing, visualization
Citations	27 in WoS, 29 in Scopus, 52 in GS

¹Available at elsevier.com/researcher/author/policies-and-guidelines/credit-author-statement (accessed on March 27, 2024).

- [4c] M. Petrlik, **P. Petracek**, V. Kratky, T. Musil, Y. Stasinchuk, M. Vrba, T. Baca, D. Hert, M. Pecka, T. Svoboda, and M. Saska, “UAVs Beneath the Surface: Cooperative Autonomy for Subterranean Search and Rescue in DARPA SubT,” *Field Robotics*, vol. 3, no. 1, pp. 1–68, 2023.

JIF	N/A — <i>Field Robotics</i> is officially approved by Dean of FEE, CTU to be considered as a journal with JIF
Contributions	MP: 30%, PP: 15% , VK: 15%, TM: 10%, YS: 6%, MV: 6%, TB: 6%, DH: 6%, MP: 2%, TS: 2%, MS: 2%
CRedit	conceptualization, methodology, software, validation, formal analysis, investigation, data curation, writing, visualization
Citations	20 in GS

- [5c] **P. Petracek**, V. Walter, T. Baca, and M. Saska, “Bio-Inspired Compact Swarms of Unmanned Aerial Vehicles without Communication and External Localization,” *Bioinspiration & Biomimetics*, vol. 16, no. 2, p. 026 009, 2021.

JIF	3.0 (Q2 in Engineering, Multidisciplinary)
Contributions	PP: 70% , VW: 20%, TB: 5%, MS: 5%
CRedit	conceptualization, methodology, software, validation, formal analysis, investigation, writing, visualization
Citations	21 in WoS, 24 in Scopus, 39 in GS

■ A.1.2 Submitted journal articles

- [6c] **P. Petracek**, K. Alexis, and M. Saska, *RMS: Redundancy-Minimizing Point Cloud Sampling for Real-Time Pose Estimation*, 2023. **Submitted to IEEE Robotics and Automation Letters (first decision Revise and Resubmit received on Jan 29, 2024).**

Contributions	PP: 90% , KA: 5%, MS: 5%
CRedit	conceptualization, methodology, software, validation, formal analysis, investigation, writing, visualization
Citations	0 in GS

■ A.2 Other peer-reviewed publications

■ A.2.1 Journal articles

- [7a] K. Ebadi, L. Bernreiter, H. Biggie, G. Catt, Y. Chang, A. Chatterjee, C. E. Denniston, S.-P. Deschenes, K. Harlow, S. Khattak, L. Nogueira, M. Palieri, **P. Petracek**, M. Petrlik, A. Reinke, V. Kratky, S. Zhao, A.-a. Agha-mohammadi, K. Alexis, C. Heckman, K. Khosoussi, N. Kottege, B. Morrell, M. Hutter, F. Pauling, F. c. Pomerleau, M. Saska, S. Scherer, R. Siegwart, J. L. Williams, and L. Carlone, “Present and Future of SLAM in Extreme Environments: The DARPA SubT Challenge,” *IEEE Transactions on Robotics*, vol. 40, pp. 936–959, 2024.

JIF	7.8 (Q1 in Robotics)
Contributions	PP: 3%
CRedit	software, validation, investigation, writing, visualization
Citations	3 in WoS, 4 in Scopus, 65 in GS

- [8a] V. Kratky, **P. Petracek**, T. Baca, and M. Saska, “An Autonomous Unmanned Aerial Vehicle System for Fast Exploration of Large Complex Indoor Environments,” *Journal of Field Robotics*, vol. 38, no. 8, pp. 1036–1058, 2021.

JIF 6.4 (Q2 in Robotics)
Contributions VK: 55%, **PP: 35%**, TB: 5%, MS: 5%
CRedit conceptualization, methodology, software, validation, investigation, writing, visualization
Citations 18 in WoS, 17 in Scopus, 40 in GS

- [9a] V. Kratky, **P. Petracek**, V. Spurny, and M. Saska, “Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2302–2309, 2020.

JIF 3.7 (Q2 in Robotics)
Contributions VK: 60%, **PP: 20%**, VS: 10%, MS: 10%
CRedit software, validation, investigation, writing (review and editing), visualization
Citations 15 in WoS, 18 in Scopus, 32 in GS

- [10a] V. Kratky, **P. Petracek**, T. Nascimento, M. Cadilova, M. Skobrtal, P. Stoudek, and M. Saska, “Safe Documentation of Historical Monuments by an Autonomous Unmanned Aerial Vehicle,” *ISPRS International Journal of Geo-Information*, vol. 10, no. 11, p. 738, 2021. **Equal contribution of the first two co-authors.**

JIF 3.1 (Q2 in Geography, Physical)
Contributions VK: 30%, **PP: 30%**, TN: 16%, MC: 6%, MS: 6%, PS: 6%, MS: 6%
CRedit conceptualization, methodology, software, validation, formal analysis, investigation, writing, visualization, project administration
Citations 5 in WoS, 4 in Scopus, 12 in GS

- [11a] F. Novak, V. Walter, **P. Petracek**, T. Baca, and M. Saska, “Fast Collective Evasion in Self-Localized Swarms of Unmanned Aerial Vehicles,” *Bioinspiration & Biomimetics*, vol. 16, no. 6, p. 066025, 2021.

JIF 3.0 (Q2 in Engineering, Multidisciplinary)
Contributions FN: 80%, VW: 5%, **PP: 5%**, TB: 5%, MS: 5%
CRedit investigation, writing (review and editing)
Citations 9 in WoS, 8 in Scopus, 13 in GS

- [12a] T. Roucek, M. Pecka, P. Cizek, T. Petricek, J. Bayer, V. Salansky, T. Azayev, D. Hert, M. Petrlik, T. Baca, V. Spurny, V. Kratky, **P. Petracek**, D. Baril, M. Vaidis, V. Kubelka, F. Pomerleau, J. Faigl, K. Zimmermann, M. Saska, T. Svoboda, and T. Krajnik, “System for Multi-Robotic Exploration of Underground Environments CTU-CRAS-NORLAB in the DARPA Subterranean Challenge,” *Field Robotics*, vol. 2, pp. 1779–1818, 2022.

JIF N/A — *Field Robotics* is officially approved by Dean of FEE, CTU to be considered as a journal with JIF
Contributions **PP: 4%**
CRedit methodology, software, validation, investigation, writing
Citations 43 in GS

- [13a] D. Hert, T. Baca, **P. Petracek**, V. Kratky, R. Penicka, V. Spurny, M. Petrlik, M. Vrba, D. Zaitlik, P. Stoudek, V. Walter, P. Stepan, J. Horyna, V. Pritzl, M. Sramek, A. Ahmad, G. Silano, D. B. Licea, P. Stibinger, T. Nascimento, and M. Saska, “MRS Drone: A Modular Platform for Real-World Deployment of Aerial Multi-Robot Systems,” *Journal of Intelligent & Robotic Systems*, vol. 108, no. 4, p. 64, 2023.

JIF 3.3 (Q3 in Robotics)
Contributions **PP: 5%**
CRedit methodology, software, validation, investigation, writing
Citations 2 in WoS, 4 in Scopus, 8 in GS

- [14a] T. Manoni, D. Albani, J. Horyna, **P. Petracek**, M. Saska, and E. Ferrante, “Adaptive Arbitration of Aerial Swarm Interactions through a Gaussian Kernel for Coherent Group Motion,” *Frontiers in Robotics and AI*, vol. 9, 2022.

JIF 3.4 (Q2 in Robotics — categorized in Emerging Sources Citation Index (ESCI))
Contributions MT: 30%, AD: 30%, JH: 10%, **PP: 10%**, MS: 10%, EF: 10%
CRedit validation, investigation, writing (review and editing)
Citations 0 in WoS, 0 in Scopus, 0 in GS

■ A.2.2 Conference proceedings in WoS

- [15a] A. Ahmad, V. Walter, **P. Petracek**, M. Petrlik, T. Baca, D. Zaitlik, and M. Saska, “Autonomous Aerial Swarming in GNSS-denied Environments with High Obstacle Density,” in *IEEE International Conference on Robotics and Automation*, 2021, pp. 570–576.

Contributions AA: 60%, VW: 7%, **PP: 7%**, MP: 7%, TB: 7%, DZ: 6%, MS: 6%
CRedit validation, investigation, writing (review and editing)
Citations 10 in WoS, 16 in Scopus, 27 in GS

- [16a] T. Amorim, T. Nascimento, **P. Petracek**, G. de Masi, E. Ferrante, and M. Saska, “Self-Organized UAV Flocking Based on Proximal Control,” in *International Conference on Unmanned Aircraft Systems*, 2021, pp. 1374–1382.

Contributions TA: 45%, TN: 20%, **PP: 20%**, GM: 5%, EF: 5%, MS: 5%
CRedit software, validation, investigation, writing (review and editing)
Citations 9 in Scopus, 13 in GS

- [17a] D. Hert, T. Baca, **P. Petracek**, V. Kratky, V. Spurny, M. Petrlik, M. Vrba, D. Zaitlik, P. Stoudek, V. Walter, P. Stepan, J. Horyna, V. Pritzl, G. Silano, D. Bonilla Licea, P. Stibinger, R. Penicka, T. Nascimento, and M. Saska, “MRS Modular UAV Hardware Platforms for Supporting Research in Real-World Outdoor and Indoor Environments,” in *International Conference on Unmanned Aircraft Systems*, 2022, pp. 1264–1273.

Contributions **PP: 5%**
CRedit methodology, software, validation, investigation, writing
Citations 5 in WoS, 7 in Scopus, 30 in GS

■ A.3 Thesis-unrelated publications

- [18a] M. Skobrtal, A. Skobrtal Zlamalova, V. Kratky, and **P. Petracek**, “Průzkum malby Zvestovani Panny Marie v poutním chrámu ve Staré Vodě u Libavě pomocí bezpilotní helikoptery,” *Pamatky*, vol. 1, no. 4, pp. 16–33, 2023.

Contributions **PP: 10%**
CRedit methodology, validation, investigation
Citations 0 in GS

- [19a] T. Prihodova, G. Silano, A. Ahmad, V. Kratky, T. Baca, **P. Petracek**, V. Saskova, J. Bednar, and M. Saska, “2022 IEEE Robotics and Automation Society Summer School on Multi-Robot Systems in Prague [Education],” *IEEE Robotics and Automation Magazine*, vol. 30, no. 1, pp. 104–106, 2023.

JIF 5.7 (Q2 in Robotics)
Contributions **PP: 11%**
CRedit conceptualization, methodology, software, validation, investigation, writing (review and editing), visualization
Citations 0 in WoS, 0 in Scopus, 0 in GS

■ B Citations of Authors' Publications

Table B.1 summarizes all citations of the author's peer-reviewed publications. The citation data were gathered on March 28, 2024 using the three primary citation databases — WoS, Scopus, and GS. The citations range throughout the author's doctoral studies between the years 2019–2024.

Platform	H-index	Citations count		
		total	excl. 1 st -order SF	excl. 2 nd -order SF
Web of Science	7	138	105	65
Scopus	8	164	130	82
Google Scholar	13	450	N/A	N/A

Table B.1: Citations of the author's publications. Citations are included with and without self-citations (SFs) of the first (excluding SF of the author) and second order (excluding SF of all authors).

■ B.1 List of Citations

The citations of individual authors' publications are listed below. The listed citations exclude self-citations of both first and second order. The citations were exported from the WoS database on March 27, 2024.

- [1c] **P. Petracek**, V. Kratky, and M. Saska, "Dronument: System for Reliable Deployment of Micro Aerial Vehicles in Dark Areas of Large Historical Monuments," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2078–2085, 2020
- B. Yang, E. Yang, L. Yu, and C. Niu, "Ultrasonic- and IMU-Based High-Precision UAV Localization for the Low-Cost Autonomous Inspection in Oil and Gas Pressure Vessels," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 10, pp. 10 523–10 534, 2023.
 - L. Kovanic, B. Topitzer, P. Petovsky, P. Blistan, M. B. Gergelova, and M. Blistanova, "Review of Photogrammetric and Lidar Applications of UAV," *Applied Sciences*, vol. 13, no. 11, 2023.
 - N. Bu, J. Ge, J. Yang, and H. Ru, "Emergency Landing System of Rotor UAV in Complex Ground Environment," in *International Conference on Autonomous Unmanned Systems*, vol. 1010, 2023, pp. 2954–2964.
 - B. Yang, E. Yang, L. Yu, and C. Niu, "Adaptive Extended Kalman Filter-Based Fusion Approach for High-Precision UAV Positioning in Extremely Confined Environments," *IEEE/ASME Transactions on Mechatronics*, vol. 28, no. 1, pp. 543–554, 2023.
 - M. Jacquet, M. Kivits, H. Das, and A. Franchi, "Motor-Level N-MPC for Cooperative Active Perception With Multiple Heterogeneous UAVs," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2063–2070, 2022.
 - M. Jacquet and A. Franchi, "Enforcing Vision-Based Localization using Perception Constrained N-MPC for Multi-Rotor Aerial Vehicles," in *IEEE International Conference on Intelligent Robots and Systems*, 2022, pp. 1818–1824.
 - G. Corsini, M. Jacquet, H. Das, A. Affi, D. Sidobre, and A. Franchi, "Nonlinear Model Predictive Control for Human-Robot Handover with Application to the Aerial Case," in *IEEE International Conference on Intelligent Robots and Systems*, 2022, pp. 7597–7604.

- M. Niarchos, M. E. Stamatiadou, C. Dimoulas, A. Veglis, and A. Symeonidis, “A Semantic Preprocessing Framework for Breaking News Detection to Support Future Drone Journalism Services,” *Future Internet*, vol. 14, no. 1, 2022.
 - A. Alcantara, J. Capitan, A. Torres-Gonzalez, R. Cunha, and A. Ollero, “Autonomous Execution of Cinematographic Shots With Multiple Drones,” *IEEE Access*, vol. 8, pp. 201 300–201 316, 2020.
- [3c] **P. Petracek**, V. Kratky, M. Petrlik, T. Baca, R. Kratochvil, and M. Saska, “Large-Scale Exploration of Cave Environments by Unmanned Aerial Vehicles,” *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7596–7603, 2021
- P. Kuwalek, “Decomposition by Approximation With Pulse Waves Allowing Further Research on Sources of Voltage Fluctuations,” *IEEE Transactions on Industrial Electronics*, vol. 71, no. 5, pp. 5263–5273, 2024.
 - B. Zhou, W. Liu, and H. Yang, “Unmanned aerial vehicle service network design for urban monitoring,” *Transportation Research Part C: Emerging Technologies*, vol. 157, 2023.
 - P. Trybala, J. Szrek, F. Remondino, P. Kujawa, J. Wodecki, J. Blachowski, and R. Zimroz, “MIN3D Dataset: Multi-sensor 3D Mapping with an Unmanned Ground Vehicle,” *Journal of Photogrammetry Remote Sensing and Geoinformation Science*, vol. 91, no. 6, pp. 425–442, 2023.
 - M. Chen, W. Yan, Y. Feng, S. Wang, and Q. Liang, “Large-Scale Underground Mine Positioning and Mapping with LiDAR-Based Semantic Intersection Detection,” *Mining Metallurgy & Exploration*, 2023.
 - J. Yu, H. Shen, J. Xu, and T. Zhang, “ECHO: An Efficient Heuristic Viewpoint Determination Method on Frontier-Based Autonomous Exploration for Quadrotors,” *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 5047–5054, 2023.
 - M. Lyu, Y. Zhao, C. Huang, and H. Huang, “Unmanned Aerial Vehicles for Search and Rescue: A Survey,” *Remote Sensing*, vol. 15, no. 13, 2023.
 - J. Yan, X. Lin, Z. Ren, S. Zhao, J. Yu, C. Cao, P. Yin, J. Zhang, and S. Scherer, “MULTARE: Cooperative Multi-Agent Exploration With Unknown Initial Position,” *IEEE Robotics and Automation Letters*, vol. 8, no. 7, pp. 4299–4306, 2023.
 - L. Long, G. Peiheng, Z. Yunlong, P. Qi, Y. Dequan, and T. Jiaying, “Research on UAV transmission line tracking and obstacle crossing technology based on RTK high-precision positioning,” *Reviews of Adhesion and Adhesives*, vol. 11, no. 2, pp. 173–198, 2023.
 - T. Gao, R. Weng, T. Wu, R. Zhang, C. Han, X. Ji, and M. Liu, “Near-ground trajectory planning for UAVs via multi-resolution hybrid voxel-surfel map,” *Science China Technological Sciences*, vol. 66, no. 5, SI, pp. 1245–1254, 2023.
 - B. Zhou, H. Xu, and S. Shen, “RACER: Rapid Collaborative Exploration With a Decentralized Multi-UAV System,” *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 1816–1835, 2023.
 - H. Cao, L. Han, D. Li, Q. Hu, and P. Hao, “Fully Distributed Dynamic Event-triggering Formation Control of UAV Swarms under DoS Attacks,” in *IEEE Conference on Decision and Control*, 2023, pp. 3166–3173.
 - L. Chen, Y. Liu, P. Dong, J. Liang, and A. Wang, “An Intelligent Navigation Control Approach for Autonomous Unmanned Vehicles via Deep Learning-Enhanced Visual SLAM Framework,” *IEEE Access*, vol. 11, pp. 119 067–119 077, 2023.
 - H. Azpurua, M. Saboia, G. M. Freitas, L. Clark, A.-a. Agha-mohammadi, G. Pessin, M. F. M. Campos, and D. G. Macharet, “A Survey on the autonomous Exploration of confined subterranean spaces: Perspectives from real-world and industrial robotic deployments,” *Robotics and Autonomous Systems*, vol. 160, 2023.
 - T. Yu, B. Deng, J. Gui, X. Zhu, and W. Yao, “Efficient Informative Path Planning via Normalized Utility in Unknown Environments Exploration,” *Sensors*, vol. 22, no. 21, 2022.
 - M. Zhao, H. Lu, S. Cheng, S. Yang, and Y. Shi, “A multi-robot cooperative Exploration algorithm considering working efficiency and working load,” *Applied Soft Computing*, vol. 128, 2022.
 - H. Xu, Y. Zhang, B. Zhou, L. Wang, X. Yao, G. Meng, and S. Shen, “Omni-Swarm: A Decentralized Omnidirectional Visual-Inertial-UWB State Estimation System for Aerial Swarms,” *IEEE Transactions on Robotics*, vol. 38, no. 6, pp. 3374–3394, 2022.

- B. Lindqvist, S. Karlsson, A. Koval, I. Tevetzidis, J. Haluska, C. Kanellakis, A.-a. Agha-mohammadi, and G. Nikolakopoulos, “Multimodality Robotic Systems: Integrated combined legged-aerial mobility for subterranean search-and-rescue,” *Robotics and Autonomous Systems*, vol. 154, 2022.
- B. Bendris and J. Cayero Becerra, “Design and Experimental Evaluation of an Aerial Solution for Visual Inspection of Tunnel-like Infrastructures,” *Remote Sensing*, vol. 14, no. 1, 2022.

[5c] **P. Petracek**, V. Walter, T. Baca, and M. Saska, “Bio-Inspired Compact Swarms of Unmanned Aerial Vehicles without Communication and External Localization,” *Bioinspiration & Biomimetics*, vol. 16, no. 2, p. 026 009, 2021

- H. Xiong, Y. Ding, and J. Liu, “Compact and ordered swarms of unmanned aerial vehicles in cluttered environments,” *Bioinspiration & Biomimetics*, vol. 18, no. 5, 2023.
- J. Guo, J. Qi, M. Wang, C. Wu, Y. Ping, S. Li, and J. Jin, “Distributed cooperative obstacle avoidance and formation reconfiguration for multiple quadrotors: Theory and experiment,” *Aerospace Science and Technology*, vol. 136, 2023.
- J. Guo, J. Qi, M. Wang, C. Wu, and G. Yang, “Collision-Free Distributed Control for Multiple Quadrotors in Cluttered Environments With Static and Dynamic Obstacles,” *IEEE Robotics and Automation Letters*, vol. 8, no. 3, pp. 1501–1508, 2023.
- X. Wang, F. Wang, Z. Nie, Y. Ai, and T. Hu, “optiSwarm: Optical Swarm Robots Using Implicit Cooperation,” *IEEE Sensors*, vol. 22, no. 24, pp. 24 380–24 394, 2022.
- E. Nebe, M. L. Sanni, R. A. Adetona, B. O. Akinyemi, S. A. Bello, and G. A. Aderounmu, “Chaos Detection and Mitigation in Swarm of Drones using Machine Learning Techniques and Chaotic Attractors,” *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 6, pp. 449–460, 2022.
- E. Soria, “Swarms of flying robots in unknown environments,” *Science Robotics*, vol. 7, no. 66, 2022.
- R. Ourari, K. Cui, A. Elshamhory, and H. Koepl, “Nearest-Neighbor-based Collision Avoidance for Quadrotors via Reinforcement Learning,” in *IEEE International Conference on Robotics and Automation*, 2022.
- V. K. Kaliappan, T. A. Nguyen, D. Min, J.-W. Lee, and U. Sakthi, “Steering Behavior-based Multiple RUAV Obstacle Avoidance Control,” *Intelligent Automation and Soft Computing*, vol. 34, no. 1, pp. 575–591, 2022.
- F. Schilling, E. Soria, and D. Floreano, “On the Scalability of Vision-Based Drone Swarms in the Presence of Occlusions,” *IEEE Access*, vol. 10, pp. 28 133–28 146, 2022.
- E. Soria, F. Schiano, and D. Floreano, “Distributed Predictive Drone Swarms in Cluttered Environments,” *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 73–80, 2022.
- Z. Sun, “Robot Swarm Navigation: Methods, Analysis, and Applications,” in *International Conference on Big Data and Artificial Intelligence and Software Engineering*, 2021, pp. 666+.

[7a] K. Ebadi, L. Bernreiter, H. Biggie, G. Catt, Y. Chang, A. Chatterjee, C. E. Denniston, S.-P. Deschenes, K. Harlow, S. Khattak, L. Nogueira, M. Palieri, **P. Petracek**, M. Petrlik, A. Reinke, V. Kratky, S. Zhao, A.-a. Agha-mohammadi, K. Alexis, C. Heckman, K. Khosoussi, N. Kottege, B. Morrell, M. Hutter, F. Pauling, F. c. Pomerleau, M. Saska, S. Scherer, R. Siegwart, J. L. Williams, and L. Carlone, “Present and Future of SLAM in Extreme Environments: The DARPA SubT Challenge,” *IEEE Transactions on Robotics*, vol. 40, pp. 936–959, 2024

- T. Wen, Y. Fang, B. Lu, X. Zhang, and C. Tang, “LIVER: A Tightly Coupled LiDAR-Inertial-Visual State Estimator With High Robustness for Underground Environments,” *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2399–2406, 2024.
- D. Lee, M. Jung, W. Yang, and A. Kim, “LiDAR odometry survey: recent advancements and remaining challenges,” *Intelligent Service Robotics*, 2024.

- J. Laconte, D. Lisus, and T. D. Barfoot, “Toward Certifying Maps for Safe Registration-Based Localization Under Adverse Conditions,” *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1572–1579, 2024.

[8a] V. Kratky, **P. Petracek**, T. Baca, and M. Saska, “An Autonomous Unmanned Aerial Vehicle System for Fast Exploration of Large Complex Indoor Environments,” *Journal of Field Robotics*, vol. 38, no. 8, pp. 1036–1058, 2021

- G. Best, R. Garg, J. Keller, G. A. Hollinger, and S. Scherer, “Multi-robot, multi-sensor Exploration of multifarious environments with full mission aerial autonomy,” *International Journal of Robotics Research*, vol. 43, no. 4, SI, pp. 485–512, 2024.
- P. Trybala, J. Szrek, F. Remondino, P. Kujawa, J. Wodecki, J. Blachowski, and R. Zimroz, “MIN3D Dataset: Multi-sensor 3D Mapping with an Unmanned Ground Vehicle,” *Journal of Photogrammetry Remote Sensing and Geoinformation Science*, vol. 91, no. 6, pp. 425–442, 2023.
- J. Lee, J. Lim, S. Pyo, and J. Lee, “Aerial online mapping on-board system by real-time object detection for UGV path generation in unstructured outdoor environments,” *Journal of Field Robotics*, vol. 40, no. 7, pp. 1754–1765, 2023.
- H. Azpurua, M. Saboia, G. M. Freitas, L. Clark, A.-a. Agha-mohammadi, G. Pessin, M. F. M. Campos, and D. G. Macharet, “A Survey on the autonomous Exploration of confined subterranean spaces: Perspectives from real-world and industrial robotic deployments,” *Robotics and Autonomous Systems*, vol. 160, 2023.
- Y. Chang, K. Ebadi, C. E. Denniston, M. F. Ginting, A. Rosinol, A. Reinke, M. Palieri, J. Shi, A. Chatterjee, B. Morrell, A.-a. Agha-mohammadi, and L. Carlone, “LAMP 2.0: A Robust Multi-Robot SLAM System for Operation in Challenging Large-Scale Underground Environments,” *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9175–9182, 2022.
- B. Lindqvist, C. Kanellakis, S. S. Mansouri, A.-a. Agha-mohammadi, and G. Nikolakopoulos, “COMPRA: A COMPact Reactive Autonomy Framework for Subterranean MAV Based Search-And-Rescue Operations,” *Journal of Intelligent & Robotic Systems*, vol. 105, no. 3, 2022.
- R. Anand, H. M. B. Kumar, A. Raghavan, R. Maddara, and P. Anand, “Automated UAV to Survey and Monitor Ionising Radiation Levels in a Closed Environment,” *Power Electronics and Drives*, vol. 7, no. 1, pp. 134–145, 2022.
- G. Best, R. Garg, J. Keller, G. A. Hollinger, and S. Scherer, “Resilient Multi-Sensor Exploration of Multifarious Environments with a Team of Aerial Robots,” in *Robotics: Science and System*, 2022.
- D. Zhu, T. Zhou, J. Lin, Y. Fang, and M. Q.-H. Meng, “Online State-Time Trajectory Planning Using Timed-ESDF in Highly Dynamic Environments,” in *IEEE International Conference on Robotics and Automation*, 2022, pp. 3949–3955.

[9a] V. Kratky, **P. Petracek**, V. Spurny, and M. Saska, “Autonomous Reflectance Transformation Imaging by a Team of Unmanned Aerial Vehicles,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2302–2309, 2020

- Y. Castro, G. Le Goic, H. Chatoux, L. De Luca, and A. Mansouri, “A new pixel-wise data processing method for reflectance transformation imaging,” *Visual Computer*, 2023.
- J. Deckerova, K. Kucerova, and J. Faigl, “On Improvement Heuristic to Solutions of the Close Enough Traveling Salesman Problem in Environments with Obstacles,” in *European Conference on Mobile Robots*, 2023, pp. 180–185.
- A. Jaspé-Villanueva, M. Ahsan, R. Pintus, A. Giachetti, F. Marton, and E. Gobbetti, “Web-based Exploration of Annotated Multi-Layered Relightable Image Models,” *ACM Journal on Computing and Cultural Heritage*, vol. 14, no. 2, SI, 2021.
- A. Alcantara, J. Capitan, R. Cunha, and A. Ollero, “Optimal trajectory planning for cinematography with multiple Unmanned Aerial Vehicles,” *Robotics and Autonomous Systems*, vol. 140, 2021.

- Y. Tang, L. Huang, J. Dai, and H. Huang, “Edge detection of Reflectance Transformation Imaging for colorless traces,” in *Optical Sensing and Imaging Technology*, vol. 12065, 2021.
- [10a] V. Kratky, **P. Petracek**, T. Nascimento, M. Cadilova, M. Skobrtal, P. Stoudek, and M. Saska, “Safe Documentation of Historical Monuments by an Autonomous Unmanned Aerial Vehicle,” *ISPRS International Journal of Geo-Information*, vol. 10, no. 11, p. 738, 2021. **Equal contribution of the first two co-authors**
- L. Kovanic, B. Topitzer, P. Petovsky, P. Blistan, M. B. Gergelova, and M. Blistanova, “Review of Photogrammetric and Lidar Applications of UAV,” *Applied Sciences*, vol. 13, no. 11, 2023.
 - A. Alsayed and M. R. A. Nabawy, “Indoor Stockpile Reconstruction Using Drone-Borne Actuated Single-Point LiDARs,” *Drones*, vol. 6, no. 12, 2022.
- [11a] F. Novak, V. Walter, **P. Petracek**, T. Baca, and M. Saska, “Fast Collective Evasion in Self-Localized Swarms of Unmanned Aerial Vehicles,” *Bioinspiration & Biomimetics*, vol. 16, no. 6, p. 066 025, 2021
- J. Li and S. X. Yang, “Intelligent Collective Escape of Swarm Robots Based on a Novel Fish-Inspired Self-Adaptive Approach With Neurodynamic Models,” *IEEE Transactions on Industrial Electronics*, 2024.
 - —, “Intelligent Fish-Inspired Foraging of Swarm Robots with Sub-Group Behaviors Based on Neurodynamic Models,” *Biomimetics*, vol. 9, no. 1, 2024.
 - L. Kong, L. Wang, Z. Cao, and X. Wang, “Resilience evaluation of UAV swarm considering resource supplementation,” *Reliability Engineering & System Safety*, vol. 241, 2024.
- [15a] A. Ahmad, V. Walter, **P. Petracek**, M. Petrlik, T. Baca, D. Zaitlik, and M. Saska, “Autonomous Aerial Swarming in GNSS-denied Environments with High Obstacle Density,” in *IEEE International Conference on Robotics and Automation*, 2021, pp. 570–576
- Z. Zhou, C. Ouyang, L. Hu, Y. Xie, Y. Chen, and Z. Gan, “A framework for dynamical distributed flocking control in dense environments,” *Expert Systems with Applications*, vol. 241, 2024.
 - X. Xu, Z. Sheng, C. Li, C. Ning, L. Zhong, B. Lin, X. Yang, and X. Zhang, “A vision-only relative distance calculation method for multi-UAV systems,” *Aerospace Science and Technology*, vol. 142, no. B, 2023.
 - J. F. Ferreira, D. Portugal, M. E. Andrada, P. Machado, R. P. Rocha, and P. Peixoto, “Sensing and Artificial Perception for Robots in Precision Forestry: A Survey,” *Robotics*, vol. 12, no. 5, 2023.
 - L. Bartolomei, L. Teixeira, and M. Chli, “Fast Multi-UAV Decentralized Exploration of Forests,” *IEEE Robotics and Automation Letters*, vol. 8, no. 9, pp. 5576–5583, 2023.
 - J. L. Lourenco, L. C. Bento, A. P. Coimbra, and A. T. De Almeida, “Comparative Evaluation of Mobile Platforms for Non-Structured Environments and Performance Requirements Identification for Forest Clearing Applications,” *Forests*, vol. 13, no. 11, 2022.
 - R. Ge, M. Lee, V. Radhakrishnan, Y. Zhou, G. Li, and G. Loianno, “Vision-based Relative Detection and Tracking for Teams of Micro Aerial Vehicles,” in *IEEE International Conference on Intelligent Robots and Systems*, 2022, pp. 380–387.

■ C References

- [20] P. Fankhauser and M. Hutter, “ANYmal: A Unique Quadruped Robot Conquering Harsh Environments,” 4 p. 2018.
- [21] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, “Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age,” *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.
- [22] N. Keetha, A. Mishra, J. Karhade, K. M. Jatavallabhula, S. Scherer, M. Krishna, and S. Garg, “AnyLoc: Towards Universal Visual Place Recognition,” *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1286–1293, 2024.
- [23] T. Musil, M. Petrlik, and M. Saska, “SphereMap: Dynamic Multi-Layer Graph Structure for Rapid Safety-Aware UAV Planning,” *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 007–11 014, 2022.
- [24] F. Blochliger, M. Fehr, M. Dymczyk, T. Schneider, and R. Siegwart, “Topomap: Topological Mapping and Navigation Based on Visual SLAM Maps,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 3818–3825.
- [25] A. Rosinol, M. Abate, Y. Chang, and L. Carlone, “Kimera: An Open-Source Library for Real-Time Metric-Semantic Localization and Mapping,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 1689–1696.
- [26] M. Grinvald, F. Furrer, T. Novkovic, J. J. Chung, C. Cadena, R. Siegwart, and J. Nieto, “Volumetric Instance-Aware Semantic Mapping and 3D Object Discovery,” *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 3037–3044, 2019.
- [27] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, “OctoMap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees,” *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, 2013.
- [28] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, “Voxblox: Incremental 3D Euclidean Signed Distance Fields for on-Board MAV Planning,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 1366–1373.
- [29] S. Li, T. Fischer, L. Ke, H. Ding, M. Danelljan, and F. Yu, “OVTrack: Open-Vocabulary Multiple Object Tracking,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 5567–5577.
- [30] B. He, H. Li, S. Wu, D. Wang, Z. Zhang, Q. Dong, C. Xu, and F. Gao, “FAST-Dynamic-Vision: Detection and Tracking Dynamic Objects with Event and Depth Sensing,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 3071–3078.
- [31] M. Tranzatto, T. Miki, M. Dharmadhikari, *et al.*, “CERBERUS in the DARPA Subterranean Challenge,” *Science Robotics*, vol. 7, no. 66, eabp9742, 2022.
- [32] T. Ozaslan, G. Loianno, J. Keller, C. J. Taylor, V. Kumar, J. M. Wozencraft, and T. Hood, “Autonomous Navigation and Mapping for Inspection of Penstocks and Tunnels with MAVs,” *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1740–1747, 2017.
- [33] M. Nieuwenhuisen, J. Quenzel, M. Beul, D. Droschel, S. Houben, and S. Behnke, “ChimneySpector: Autonomous MAV-based Indoor Chimney Inspection Employing 3D Laser Localization and Textured Surface Reconstruction,” in *International Conference on Unmanned Aircraft Systems*, 2017, pp. 278–285.
- [34] A. Alcantara, J. Capitan, R. Cunha, and A. Ollero, “Optimal Trajectory Planning for Cinematography with Multiple Unmanned Aerial Vehicles,” *Robotics and Autonomous Systems*, vol. 140, p. 103 778, 2021.

- [35] V. Kratky, A. Alcantara, J. Capitan, P. Stepan, M. Saska, and A. Ollero, “Autonomous Aerial Filming with Distributed Lighting by a Team of Unmanned Aerial Vehicles,” *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7580–7587, 2021.
- [36] V. Trianni, *Evolutionary Swarm Robotics: Evolving Self-Organising Behaviours in Groups of Autonomous Robots*, 1st ed. 2008.
- [37] V. Walter, N. Staub, M. Saska, and A. Franchi, “Mutual Localization of UAVs Based on Blinking Ultraviolet Markers and 3D Time-Position Hough Transform,” in *IEEE CASE*, 2018, pp. 298–303.
- [38] V. Walter, N. Staub, A. Franchi, and M. Saska, “UVDAR System for Visual Relative Localization with Application to Leader–Follower Formations of Multirotor UAVs,” *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2637–2644, 2019.
- [39] T. Baca, M. Petrlik, M. Vrba, V. Spurny, R. Penicka, D. Hert, and M. Saska, “The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-World Deployment, and Education with Autonomous Unmanned Aerial Vehicles,” *Journal of Intelligent & Robotic Systems*, vol. 102, no. 1, pp. 1–28, 2021.
- [40] R. Kummerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, “G2o: A General Framework for Graph Optimization,” in *IEEE International Conference on Robotics and Automation*, 2011, pp. 3607–3613.
- [41] F. Dellaert and M. Kaess, *Factor Graphs for Robot Perception*. 2017.
- [42] J. Zhang and S. Singh, “LOAM: Lidar Odometry and Mapping in Real-Time.,” in *Robotics: Science and Systems*, vol. 2, 2014.
- [43] A. Geiger, P. Lenz, and R. Urtasun, “Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite,” in *Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3354–3361.
- [44] T. Shan and B. Englot, “LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain,” in *International Conference on Intelligent Robots and Systems*, 2018, pp. 4758–4765.
- [45] G. Xue, J. Wei, R. Li, and J. Cheng, “LeGO-LOAM-SC: An Improved Simultaneous Localization and Mapping Method Fusing LeGO-LOAM and Scan Context for Underground Coalmine,” *Sensors*, vol. 22, 2022.
- [46] G. Kim and A. Kim, “Scan Context: Egocentric Spatial Descriptor for Place Recognition within 3D Point Cloud Map,” in *International Conference on Intelligent Robots and Systems*, 2018, pp. 4802–4809.
- [47] X. Liu, L. Zhang, S. Qin, D. Tian, S. W. Ouyang, and C. Chen, “Optimized LOAM Using Ground Plane Constraints and SegMatch-Based Loop Detection,” *Sensors*, vol. 19, 2019.
- [48] Z. Liu and F. Zhang, “BALM: Bundle Adjustment for Lidar Mapping,” *IEEE Robotics and Automation Letters*, vol. 6, pp. 3184–3191, 2021.
- [49] H. Wang, C. Wang, and L. Xie, “Intensity-SLAM: Intensity Assisted Localization and Mapping for Large Scale Environment,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1715–1721, 2021.
- [50] C. Qu, S. S. Shivakumar, W. Liu, and C. J. Taylor, “LLOL: Low-Latency Odometry for Spinning Lidars,” in *IEEE International Conference on Robotics and Automation*, 2022, pp. 4149–4155.
- [51] W. Xu and F. Zhang, “FAST-LIO: A Fast, Robust LiDAR-Inertial Odometry Package by Tightly-Coupled Iterated Kalman Filter,” *IEEE Robotics and Automation Letters*, vol. 6, pp. 3317–3324, 2021.
- [52] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, “KISS-ICP: In Defense of Point-to-Point ICP – Simple, Accurate, and Robust Registration If Done the Right Way,” *IEEE Robotics and Automation Letters*, vol. 8, no. 2, pp. 1029–1036, 2023.

- [53] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, “FAST-LIO2: Fast Direct LiDAR-Inertial Odometry,” *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, 2022.
- [54] D. He, W. Xu, N. Chen, F. Kong, C. Yuan, and F. Zhang, “Point-LIO: Robust High-Bandwidth Light Detection and Ranging Inertial Odometry,” *Advanced Intelligent Systems*, vol. 5, no. 7, p. 2200459, 2023.
- [55] J. Behley and C. Stachniss, “Efficient Surfel-Based SLAM Using 3D Laser Range Data in Urban Environments,” in *Robotics: Science and Systems*, vol. 2018, 2018.
- [56] X. Chen, A. Milioto, E. Palazzolo, P. Giguère, J. Behley, and C. Stachniss, “SuMa++: Efficient LiDAR-based Semantic SLAM,” in *International Conference on Intelligent Robots and Systems*, 2019, pp. 4530–4537.
- [57] M. Yokozuka, K. Koide, S. Oishi, and A. Banno, “LiTAMIN2: Ultra Light LiDAR-based SLAM Using Geometric Approximation Applied with KL-Divergence,” *IEEE International Conference on Robotics and Automation*, pp. 11619–11625, 2021.
- [58] B. Sheng, S. Wenzhong, F. Wenzheng, C. Pengxin, N. Mingyan, and X. Haodong, “A Tight Coupling Mapping Method to Integrate the ESKF, G2o, and Point Cloud Alignment,” *The Journal of Supercomputing*, vol. 78, no. 2, pp. 1903–1922, 2021.
- [59] J. Lin and F. Zhang, “Loam Livox: A Fast, Robust, High-Precision LiDAR Odometry and Mapping Package for LiDARs of Small FoV,” in *IEEE International Conference on Robotics and Automation*, 2020, pp. 3126–3131.
- [60] M. Karimi, M. Oelsch, O. Stengel, E. Babaïans, and E. G. Steinbach, “LoLa-SLAM: Low-latency LiDAR SLAM Using Continuous Scan Slicing,” *IEEE Robotics and Automation Letters*, vol. 6, pp. 2248–2255, 2021.
- [61] H. Wang, C. Wang, and L. Xie, “Lightweight 3-D Localization and Mapping for Solid-State LiDAR,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1801–1807, 2021.
- [62] H. Ye, Y. Chen, and M. Liu, “Tightly Coupled 3D Lidar Inertial Odometry and Mapping,” *IEEE International Conference on Robotics and Automation*, pp. 3144–3150, 2019.
- [63] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, “LIO-SAM: Tightly-coupled Lidar Inertial Odometry via Smoothing and Mapping,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5135–5142, 2020.
- [64] C. Qin, H. Ye, C. E. Pranata, J. H. Han, S. Zhang, and M. Liu, “LINS: A Lidar-Inertial State Estimator for Robust and Efficient Navigation,” *IEEE International Conference on Robotics and Automation*, pp. 8899–8906, 2020.
- [65] H. Wang, C. Wang, C.-L. Chen, and L. Xie, “F-LOAM : Fast LiDAR Odometry and Mapping,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4390–4396, 2021.
- [66] Y. Su, T. Wang, S. Shao, C. Yao, and Z. Wang, “GR-LOAM: LiDAR-based Sensor Fusion SLAM for Ground Robots on Complex Terrain,” *Robotics and Autonomous Systems*, vol. 140, p. 103759, 2021.
- [67] T.-M. Nguyen, S. Yuan, M. Cao, L. Yang, T. H. Nguyen, and L. Xie, “MILIOM: Tightly Coupled Multi-Input Lidar-Inertia Odometry and Mapping,” *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5573–5580, 2021.
- [68] M. Palieri, B. Morrell, A. Thakur, K. Ebadi, J. Nash, A. Chatterjee, C. Kanellakis, L. Carlone, C. Guaragnella, and A.-a. Agha-mohammadi, “LOCUS: A Multi-Sensor Lidar-Centric Solution for High-Precision Odometry and 3D Mapping in Real-Time,” *IEEE Robotics and Automation Letters*, vol. 6, pp. 421–428, 2021.
- [69] N. Khedekar, M. Kulkarni, and K. Alexis, “MIMOSA: A Multi-Modal SLAM Framework for Resilient Autonomy against Sensor Degradation,” in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 7153–7159.
- [70] J. Zhang and S. Singh, “Enabling Aggressive Motion Estimation at Low-Drift and Accurate Mapping in Real-Time,” in *IEEE International Conference on Robotics and Automation*, 2017, pp. 5051–5058.

- [71] —, “Laser-Visual-Inertial Odometry and Mapping with High Robustness and Low Drift,” *Journal of Field Robotics*, vol. 35, no. 8, pp. 1242–1264, 2018.
- [72] T. Shan, B. Englot, C. Ratti, and D. Rus, “LVI-SAM: Tightly-coupled Lidar-Visual-Inertial Odometry via Smoothing and Mapping,” *IEEE International Conference on Robotics and Automation*, pp. 5692–5698, 2021.
- [73] J. Lin, C. Zheng, W. Xu, and F. Zhang, “R2LIVE: A Robust, Real-Time, LiDAR-Inertial-Visual Tightly-Coupled State Estimator and Mapping,” *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7469–7476, 2021.
- [74] X. Zuo, P. Geneva, W. Lee, Y. Liu, and G. Huang, “LIC-Fusion: LiDAR-Inertial-Camera Odometry,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5848–5854, 2019.
- [75] X. Zuo, Y. Yang, P. Geneva, J. Lv, Y. Liu, G. Huang, and M. Pollefeys, “LIC-Fusion 2.0: LiDAR-Inertial-Camera Odometry with Sliding-Window Plane-Feature Tracking,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5112–5119, 2020.
- [76] K. Koide, M. Yokozuka, S. Oishi, and A. Banno, “Globally Consistent 3D LiDAR Mapping with GPU-Accelerated GICP Matching Cost Factors,” *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8591–8598, 2021.
- [77] T.-M. Nguyen, S. Yuan, M. Cao, T. Nguyen, and L. Xie, “VIRAL SLAM: Tightly Coupled Camera-IMU-UWB-Lidar SLAM,” *ArXiv*, 2021.
- [78] T.-M. Nguyen, M. Cao, S. Yuan, Y. Lyu, T. H. Nguyen, and L. Xie, “LIRO: Tightly Coupled Lidar-Inertia-Ranging Odometry,” *IEEE International Conference on Robotics and Automation*, pp. 14 484–14 490, 2021.
- [79] G. Wang, S. Gao, H. Ding, H. Zhang, and H. Cai, “LIO-CSI: LiDAR Inertial Odometry with Loop Closure Combined with Semantic Information,” *PLoS ONE*, vol. 16, 2021.
- [80] S. W. Chen, G. V. Nardari, E. S. Lee, C. Qu, X. Liu, R. A. F. Romero, and V. Kumar, “SLOAM: Semantic Lidar Odometry and Mapping for Forest Inventory,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 612–619, 2020.
- [81] V. Pritzl, M. Vrba, C. Tortorici, R. Ashour, and M. Saska, “Adaptive Estimation of UAV Altitude in Complex Indoor Environments Using Degraded and Time-Delayed Measurements with Time-Varying Uncertainties,” *Robotics and Autonomous Systems*, vol. 160, p. 104 315, 2023.
- [82] J. Zhang, M. Kaess, and S. Singh, “On Degeneracy of Optimization-Based State Estimation Problems,” in *IEEE International Conference on Robotics and Automation*, 2016, pp. 809–816.
- [83] W. Zhen and S. Scherer, “Estimating the Localizability in Tunnel-like Environments Using LiDAR and UWB,” in *IEEE International Conference on Robotics and Automation*, 2019, pp. 4903–4908.
- [84] J. Jiao, Y. Zhu, H. Ye, H. Huang, P. Yun, L. Jiang, L. Wang, and M. Liu, “Greedy-Based Feature Selection for Efficient LiDAR SLAM,” *IEEE International Conference on Robotics and Automation*, pp. 5222–5228, 2021.
- [85] W. Li, Y. Hu, Y. Han, and X. Li, “KFS-LIO: Key-feature Selection for Lightweight Lidar Inertial Odometry,” *IEEE International Conference on Robotics and Automation*, pp. 5042–5048, 2021.
- [86] Y. Wang, Y. Lou, Y. Zhang, W. Song, F. Huang, and Z. Tu, “A Robust Framework for Simultaneous Localization and Mapping with Multiple Non-Repetitive Scanning Lidars,” *Remote Sensing*, vol. 13, no. 10, p. 2015, 2021.
- [87] T. Tuna, J. Nubert, Y. Nava, S. Khattak, and M. Hutter, “X-ICP: Localizability-Aware LiDAR Registration for Robust Localization in Extreme Environments,” *IEEE Transactions on Robotics*, vol. 40, pp. 452–471, 2024.
- [88] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, “PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space,” in *Advances in Neural Information Processing Systems*, vol. 30, 2017.

- [89] S. Rusinkiewicz and M. Levoy, “Efficient Variants of the ICP Algorithm,” in *3DIM*, 2001, pp. 145–152.
- [90] N. Gelfand, L. Ikemoto, S. Rusinkiewicz, and M. Levoy, “Geometrically Stable Sampling for the ICP Algorithm,” in *3DIM*, 2003, pp. 260–267.
- [91] I. Lang, A. Manor, and S. Avidan, “SampleNet: Differentiable Point Cloud Sampling,” in *IEEE/CVF CVPR*, 2020, pp. 7575–7585.
- [92] A. Kushleyev, D. Mellinger, C. Powers, and V. Kumar, “Towards a Swarm of Agile Micro Quadrotors,” *Autonomous Robots*, vol. 35, no. 4, pp. 287–300, 2013.
- [93] A. Weinstein, A. Cho, G. Loianno, and V. Kumar, “Visual Inertial Odometry Swarm: An Autonomous Swarm of Vision-Based Quadrotors,” *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1801–1807, 2018.
- [94] P. Schmuck and M. Chli, “Multi-UAV Collaborative Monocular SLAM,” in *IEEE International Conference on Robotics and Automation*, 2017, pp. 3863–3870.
- [95] —, “CCM-SLAM: Robust and Efficient Centralized Collaborative Monocular Simultaneous Localization and Mapping for Robotic Teams,” *Journal of Field Robotics*, vol. 36, pp. 763–781, 2019.
- [96] C. W. Reynolds, “Flocks, Herds and Schools: A Distributed Behavioral Model,” in *14th Ann. Conf. on Computer Graphics and Interactive Techniques*, 1987, pp. 25–34.
- [97] R. Olfati-Saber, “Flocking for Multi-Agent Dynamic Systems: Algorithms and Theory,” *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.
- [98] T. Schmickl and H. Hamann, “Beeclust: A Swarm Algorithm Derived from Honeybees Derivation of the Algorithm, Analysis by Mathematical Models and Implementation on a Robot Swarm,” in *Bio-Inspired Computing and Networking*, 2016, pp. 95–137.
- [99] M. Bodi, C. Moslinger, R. Thenius, and T. Schmickl, “BEECLUST Used for Exploration Tasks in Autonomous Underwater Vehicles,” *International Conference on Mathematical Modelling (IFAC-PapersOnLine)*, vol. 48, no. 1, pp. 819–824, 2015.
- [100] D. Shah and L. Vachhani, “Swarm Aggregation without Communication and Global Positioning,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 886–893, 2019.
- [101] I. Olaronke, R. Ikono, I. Gambo, and O. Ojerinde, “A Systematic Review of Swarm Robots,” *Current Journal of Applied Science and Technology*, vol. 39, pp. 79–97, 2020.
- [102] S.-J. Chung, A. A. Paranjape, P. M. Dames, S. Shen, and V. R. Kumar, “A Survey on Aerial Swarm Robotics,” *IEEE Transactions on Robotics*, vol. 34, pp. 837–855, 2018.
- [103] C. Viragh, G. Vasarhelyi, N. Tarcai, T. Szorenyi, G. Somorjai, T. Nepusz, and T. Vicsek, “Flocking Algorithm for Autonomous Flying Robots,” *Bioinspiration & Biomimetics*, vol. 9, no. 2, p. 025 012, 2014.
- [104] G. Vasarhelyi, e. C. Viragh, G. Somorjai, N. Tarcai, T. Szorenyi, T. Nepusz, and T. Vicsek, “Outdoor Flocking and Formation Flight with Autonomous Aerial Robots,” in *International Conference on Intelligent Robots and Systems*, 2014, pp. 3866–3873.
- [105] G. Vasarhelyi, C. Viragh, G. Somorjai, T. Nepusz, A. E. Eiben, and T. Vicsek, “Optimized Flocking of Autonomous Drones in Confined Environments,” *Science Robotics*, vol. 3, no. 20, eaat3536, 2018.
- [106] X. Zhou, X. Wen, Z. Wang, *et al.*, “Swarm of Micro Flying Robots in the Wild,” *Science Robotics*, vol. 7, no. 66, eabm5954, 2022.
- [107] T.-M. Nguyen, Z. Qiu, T. H. Nguyen, M. Cao, and L. Xie, “Distance-Based Cooperative Relative Localization for Leader-Following Control of MAVs,” *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3641–3648, 2019.
- [108] H.-w. Park, I.-s. Choi, S.-k. Park, and J.-s. Choi, “Leader-Follower Formation Control Using Infrared Camera with Reflective Tag,” in *International Conference on Ubiquitous Robots and Ambient Intelligence*, 2013, pp. 321–324.

- [109] A. Censi, J. Strubel, C. Brandli, T. Delbruck, and D. Scaramuzza, “Low-Latency Localization by Active LED Markers Tracking Using a Dynamic Vision Sensor,” in *International Conference on Intelligent Robots and Systems*, 2013, pp. 891–898.
- [110] O. De Silva, G. K. I. Mann, and R. G. Gosine, “An Ultrasonic and Vision-Based Relative Positioning Sensor for Multirobot Localization,” *IEEE Sensors*, vol. 15, no. 3, pp. 1716–1726, 2015.
- [111] X. Yan, H. Deng, and Q. Quan, “Active Infrared Coded Target Design and Pose Estimation for Multiple Objects,” in *International Conference on Intelligent Robots and Systems*, 2019, pp. 6885–6890.
- [112] K. Chaudhary, M. Zhao, F. Shi, X. Chen, K. Okada, and M. Inaba, “Robust Real-Time Visual Tracking Using Dual-Frame Deep Comparison Network Integrated with Correlation Filters,” in *International Conference on Intelligent Robots and Systems*, 2017, pp. 6837–6842.
- [113] A. Carrio, J. Tordesillas, S. Vemprala, S. Saripalli, P. Campoy, and J. P. How, “Onboard Detection and Localization of Drones Using Depth Maps,” *IEEE Access*, vol. 8, pp. 30 480–30 490, 2020.
- [114] M. Vrba and M. Saska, “Marker-Less Micro Aerial Vehicle Detection and Localization Using Convolutional Neural Networks,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2459–2466, 2020.
- [115] J. Faigl, T. Krajnik, J. Chudoba, L. Preucil, and M. Saska, “Low-Cost Embedded System for Relative Localization in Robotic Swarms,” in *IEEE International Conference on Robotics and Automation*, 2013, pp. 993–998.
- [116] M. Saska, T. Baca, J. Thomas, J. Chudoba, L. Preucil, T. Krajnik, J. Faigl, G. Loianno, and V. Kumar, “System for Deployment of Groups of Unmanned Micro Aerial Vehicles in GPS-denied Environments Using Onboard Visual Relative Localization,” *Autonomous Robots*, vol. 41, no. 4, pp. 919–944, 2017.
- [117] A. Dmytruk, T. Nascimento, A. Ahmad, T. Baca, and M. Saska, “Safe Tightly-Constrained UAV Swarming in GNSS-denied Environments,” in *International Conference on Unmanned Aircraft Systems*, 2021, pp. 1391–1399.
- [118] A. Ahmad, D. B. Licea, G. Silano, T. Baca, and M. Saska, “PACNav: A Collective Navigation Approach for UAV Swarms Deprived of Communication and External Localization,” *Bioinspiration & Biomimetics*, vol. 17, no. 6, p. 066 019, 2022.
- [119] S. A. Prieto, B. Quintana, A. Adan, and A. S. Vazquez, “As-Is Building-Structure Reconstruction from a Probabilistic next Best Scan Approach,” *Robotics and Autonomous Systems*, vol. 94, pp. 186–207, 2017.
- [120] M. Beul, D. Droschel, M. Nieuwenhuisen, J. Quenzel, S. Houben, and S. Behnke, “Fast Autonomous Flight in Warehouses for Inventory Applications,” *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3121–3128, 2018.
- [121] M. Ceccarelli, D. Cafolla, G. Carbone, *et al.*, “HeritageBot Service Robot Assisting in Cultural Heritage,” in *2017 First IEEE International Conference on Robotic Computing (IRC)*, 2017, pp. 440–445.
- [122] T. Baca, R. Penicka, P. Stepan, M. Petrlik, V. Spurny, D. Hert, and M. Saska, “Autonomous Cooperative Wall Building by a Team of Unmanned Aerial Vehicles in the MBZIRC 2020 Competition,” *Robotics and Autonomous Systems*, vol. 167, p. 104 482, 2023.
- [123] M. Nieuwenhuisen, M. Beul, R. A. Rosu, J. Quenzel, D. Pavlichenko, S. Houben, and S. Behnke, “Collaborative Object Picking and Delivery with a Team of Micro Aerial Vehicles at MBZIRC,” in *2017 European Conference on Mobile Robots (ECMR)*, 2017, pp. 1–6.
- [124] V. Walter, V. Spurny, M. Petrlik, T. Baca, D. Zaitlik, L. Demkiv, and M. Saska, “Extinguishing Real Fires by Fully Autonomous Multirotor UAVs in the MBZIRC 2020 Competition,” *Field Robotics*, vol. 2, no. 1, pp. 406–436, 2022.

- [125] J. Quenzel, M. Splietker, D. Pavlichenko, D. Schleich, C. Lenz, M. Schwarz, M. Schreiber, M. Beul, and S. Behnke, “Autonomous Fire Fighting with a UAV-UGV Team at MBZIRC 2020,” in *International Conference on Unmanned Aircraft Systems*, 2021, pp. 934–941.
- [126] M. Vrba, Y. Stasinchuk, T. Baca, V. Spurny, M. Petrlik, D. Hert, D. Zaitlik, and M. Saska, “Autonomous Capture of Agile Flying Objects Using UAVs: The MBZIRC 2020 Challenge,” *Robotics and Autonomous Systems*, vol. 149, p. 103970, 2022.
- [127] M. Zhao, F. Shi, T. Anzai, T. Nishio, T. Maki, K. Ito, N. Kuromiya, K. Okada, and M. Inaba, “Team JSK at MBZIRC 2020: Interception of Fast Flying Target Using Multilinked Aerial Robot,” *Field Robotics*, vol. 1, no. 1, pp. 70–101, 2021.
- [128] A. Agha, K. Otsu, B. Morrell, *et al.*, “NeBula: Quest for Robotic Autonomy in Challenging Environments; TEAM CoSTAR at the DARPA Subterranean Challenge,” *ArXiv*, 2021.
- [129] N. Hudson, F. Talbot, M. Cox, *et al.*, “Heterogeneous Ground and Air Platforms, Homogeneous Sensing: Team CSIRO Data61’s Approach to the DARPA Subterranean Challenge,” *Field Robotics*, vol. 2, no. 1, pp. 595–636, 2022.
- [130] Government Office of the Czech Republic. (2022). Evaluation of Methodology M17+, [Online]. Available: <https://m17.rvvi.cz/en/> (accessed on 28/03/2024).
- [131] M. Saska, V. Kratky, V. Spurny, and T. Baca, “Documentation of Dark Areas of Large Historical Buildings by a Formation of Unmanned Aerial Vehicles Using Model Predictive Control,” in *IEEE ETFA*, 2017, pp. 1–8.
- [132] D. Smrcka, T. Baca, T. Nascimento, and M. Saska, “Admittance Force-Based UAV-Wall Stabilization and Press Exertion for Documentation and Inspection of Historical Buildings,” in *International Conference on Unmanned Aircraft Systems*, 2021, pp. 552–559.
- [133] M. Petrlik, T. Baca, D. Hert, M. Vrba, T. Krajnik, and M. Saska, “A Robust UAV System for Operations in a Constrained Environment,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2169–2176, 2020.
- [134] T. Baca, P. Stibinger, D. Doubravova, D. Turecek, J. Solc, J. Rusnak, M. Saska, and J. Jakubek, “Gamma Radiation Source Localization for Micro Aerial Vehicles with a Miniature Single-Detector Compton Event Camera,” in *International Conference on Unmanned Aircraft Systems*, 2021, pp. 338–346.
- [135] K. Cesare, R. Skeelee, S.-H. Yoo, Y. Zhang, and G. Hollinger, “Multi-UAV Exploration with Limited Communication and Battery,” in *IEEE International Conference on Robotics and Automation*, 2015, pp. 2230–2235.
- [136] J. Horyna, T. Baca, V. Walter, D. Albani, D. Hert, E. Ferrante, and M. Saska, “Decentralized Swarms of Unmanned Aerial Vehicles for Search and Rescue Operations without Explicit Communication,” *Autonomous Robots*, vol. 47, no. 1, pp. 77–93, 2023.
- [137] Multi-Robot Systems group, CTU FEE. (2024). Dronument (Drone & Monument), [Online]. Available: <https://mrs.felk.cvut.cz/dronument> (accessed on 28/03/2024).