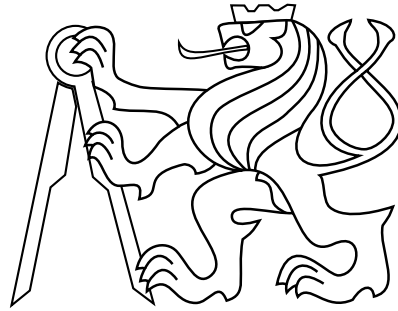CZECH TECHNICAL UNIVERSITY IN PRAGUE

FACULTY OF ELECTRICAL ENGINEERING
DEPARTMENT OF CYBERNETICS
MULTI-ROBOT SYSTEMS

Visual Relative Localization and Stabilization of Cooperating Unmanned Helicopters

Doctoral Thesis

Ing. Viktor Walter

Prague, February, 2024

Ph.D. Program: Electrical Engineering and Information Technology
Branch of Study: Artificial Intelligence and Biocybernetics

Supervisor: doc. Ing. Martin Saska, Dr. rer. nat.

Acknowledgments

I thank my supervisor Martin Saska, as well as Matouš Vrba, Vojtěch Spurný, David Žaitlík, Matej Hilmer, Daniel Bonilla Licea, Nicolas Staub, Tomáš Báča, Daniel Heřt, Matej Petrlík, Pavel Petráček and the whole Multi-robot Systems Group and all my co-authors. I am extremely grateful for their support and patience with me, and I will always cherish their camaraderie and memorable moments we had together.

I would also like to express my gratitude to Antonio Franchi and his team in LAAS where I learned valuable skills and developed some of the core parts of this thesis, as well as to all CTU-UPENN-NYU team members that competed in the MBZIRC competition.

Last, but not least, I would like to express utmost gratitude to my family and my friends who selflessly supported me throughout this period of my life.

Legal

This thesis compiles several journal articles and conference proceedings. These works are subject to copyright by the respective publishers and may not be re-published without their permission.

IEEE material
In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of CTU's products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or re-distribution, please go to `http://www.ieee.org/publications_standards/publications/rights/rights_link.html` to learn how to obtain a License from RightsLink. If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.
The relevant materials were used with permission as declared at `https://conferences.ieeeauthorcenter.ieee.org/get-published/about-transferring-copyright-to-ieee/#thesis` [accessed 9.11.2023]

IOP material
The attached full text of the manuscript in question is a preprint version of the published article, permissible as per IOP rules found at `https://publishingsupport.iopscience.iop.org/permissionsfaq/` [accessed 30.11.2023]

Abstract

Scenarios where multiple autonomous aerial vehicles are required to closely cooperate in a shared operational space require mutual relative localization. In previous instances, the requirement of relative localization was often circumvented in controlled conditions by using explicit global localization obtained from external infrastructure. Such an approach may be suitable for constrained experimental conditions for testing of theoretical cooperative behaviors. However, this solution is not suitable for practical multirobot Unmanned Aerial Vehicle (UAV) deployment into arbitrary environments where such infrastructure is unavailable without direct access by the operator. The main outcome of my work is the Vision-based Relative Localization (VRL) system UltraViolet Direction And Ranging ($UVDAR$), that exploits active ultraviolet (UV) markers and binary blinking signalization, combined with specialized computer vision processing, such that it addresses the drawbacks of prior onboard relative localization systems. This system enabled performing real-world aerial multi-robot operations without reliance on external infrastructure. The development of such a system revealed new challenges stemming from the non-ideal properties of a real relative localization system that we addressed. Additionally, we were able to exploit beneficial properties of the system unforeseen by theoretical prior works, by enabling relative orientation estimation and optical data transmission. $UVDAR$ was used in real-world swarm and formation flights, and it also allowed us to create and publish a practical annotated image dataset for detection of UAVs by Machine Learning (ML)-based computer vision. Besides the work derived from relative localization of UAVs, I am also presenting the outcomes of my successful participation in the Mohamed Bin Zayed International Robotics Challenge (MBZIRC) 2020 robotic competition, where I have developed tools used in the remainder of my work. These works primarily address the issue of stabilization and aiming with a UAV system with onboard computer vision in the loop.

Keywords: Unmanned Aerial Vehicles, Computer Vision, Visual Relative Localization, Multi-Robot Systems, Swarms

Abstrakt

Scénáře, ve kterých multirobotické letouny musí úzce spolupracovat ve sdílené operační oblasti, vyžadují vzájemnou relativní lokalizaci. Stávající realizace multirobotických systémů tento předpoklad často naplňují použitím explicitní globální lokalizace za využití externí infrastruktury. Takový přístup může být vhodný v řízených podmínkách při experimentálním ověřování teoretických algoritmů kooperativního chování, ale podobná řešení nejsou vhodná pro praktické nasazení multirobotických týmů bezpilotních letounů v libovolných prostředích, v nichž se nedá spolehnout na přítomnost nutné externí infrastruktury, nebo na možnost instalace takové infrastruktury operátorem. Hlavní výstup mé práce je systém relativní lokalizace založený na počítačovém vidění jménem UVDAR (Ultra-Violet Direction And Ranging), který využívá binárně kódované aktivní ultrafialové značky kombinované se specializovanými algoritmy zpracování obrazu, a který řeší omezení předchozích systémů palubní relativní lokalizace. Tento systém umožňuje nasazení multirobotických letounů v reálném světě bez závislosti na externí infrastruktuře. Vývoj tohoto systému odhalil nové výzvy vyvstávající z neideálních vlastností reálných systémů relativní lokalizace, které jsme vyřešili. Navíc jsme s výhodou využili žádoucí vlastnosti systému, které teoretická literatura nepředpověděla, na získání odhadu relativní orientace a na optický přenos dat. Nový systém byl použit pro přenos rojových a formačních algoritmů z teorie a simulace do experimentů v reálném světě nebo pro automatickou anotaci obrazových datasetů určených ke trénování neuronových sítí na detekci letounů. Kromě práce týkající se relativní loakalizace bezpilotních letounů prezentuji také výsledky mé úspěšné účasti na robotické soutěži Mohamed Bin Zayed International Robotics Challenge (MBZIRC) 2020, kde jsem vyvinul nástroje, které jsem následně používal ve zbytku své práce. Tyto nástroje se týkaly primárně stabilizace a míření bezpilotním letounem pomocí zpětné vazby z palubního počítačového vidění.

Klíčová slova Bezpilotní letouny, Počítačové vidění, Vizuální relativní lokalizace, Multirobotické systémy, Roje

Contents

Abbreviations

IR	Infrared
GNSS	Global Navigation Satellite System
mo-cap	Motion capture
MRS	Multi-robot Systems Group
ML	Machine Learning
UAV	Unmanned Aerial Vehicle
UV	ultraviolet
UVDAR	UltraViolet Direction And Ranging
UWB	Ultra-Wide Bandwidth
RTK	Real-Time Kinematic
ROS	Robot Operating System
FEC	formation-enforcing control
LiDAR	Light Detection and Ranging
HT	Hough Transform
MBZIRC	Mohamed Bin Zayed International Robotics Challenge
VRL	Vision-based Relative Localization
TII	Technology Innovation Institute

Chapter 1

Introduction

Technological developments of the last two decades have given rise to readily available, lightweight Unmanned Aerial Vehicles (UAVs). These devices are now capable of autonomous robotic flight [1] and they can be equipped with various onboard sensors, such as Light Detection and Ranging (LiDAR) or computer-vision enabled cameras. Thus, they show promise for a variety of applications, including surveillance [2]–[5], search and rescue [6], [7], and robotic assistance in challenging conditions [8]. Compared to robotic ground vehicles, UAVs are faster, more lightweight, and capable of bypassing difficult terrain. On the other hand, their limited carrying capacity restricts their application in terms of both the physical payload, as well as the energy and computational resources that can be carried onboard. These detriments can be mitigated by cooperation of multiple simultaneously deployed UAVs. Such cooperation can additionally provide capabilities that are otherwise unavailable to robotic systems, such as the entire system changing shape for a specific task, redundancy in the case of failure of an individual part in the system, and large-scale parallel area coverage. Effective cooperation requires that the UAV group in question is equipped with some means of mutual relative localization, and can benefit greatly if active communication between them is also available. In previous works, this was typically achieved using external infrastructure present in the deployment area. Examples of such an infrastructure include Global Navigation Satellite System (GNSS), which can only achieve the necessary level of precision for cooperation by using differential measurement, as in Real-Time Kinematic (RTK) systems [9], [10] or Motion capture (mo-cap) systems using a calibrated set of specialized cameras and markers[11].

Both systems rely on external infrastructure that must be prepared near the area of deployment. This requirement is a crucial drawback that makes these systems unsuitable for practical deployment in arbitrary locations. The infrastructure in question must be accessible within the entire operational space, meaning that, due to inaccessibility, these systems can typically not be used in the e.g. search-and-rescue operations that would benefit from using UAVs. In addition to the constraints presented by the installation of physical infrastructure, these mutual relative localization methods require communication using radio channels to transfer localization data to the UAVs. Such wireless communications are prone to congestion and interference, either circumstantial or deliberate.

Mutual relative localization methods of UAVs not requiring external infrastructure, but rather relying purely on onboard equipment, can be broadly divided into distance-based methods and Vision-based Relative Localization (VRL). The basis of a typical example of a distance-based relative localization method is Ultra-Wide Bandwidth (UWB) ranging, which is a time-of-flight ranging system using the transmission of wide-bandwidth pulses [12]. A drawback of onboard relative localization using distance sensors is that, without exhibiting specific behavior, reconstructing the entire relative pose of a neighbor is not possible [13], thus limiting the utility of this approach.

VRL, on the other hand, typically extracts the relative direction towards a target, i.e., the *bearing* of the target. It is also possible to extract clues on its relative distance, either through the appearance of the object in the image, or by using stereo vision.

One method of relative localization having the properties of both methods is based on onboard LiDAR scanners [14], though such devices are, at the time of writing, too costly for large-scale UAV cooperation.

The approach focused on in my research is VRL, which has been enabled in recent years by the increasing computational power of the lightweight, low-power computers carriable by UAVs. Prior works have leveraged passive visual markers attached to targets that can be extracted and localized by computer vision algorithms. These markers are usually in the form of simple shapes distinguished by a color assumed to be absent in the environment [15], or by more specific patterns containing identifying information [16]–[18]. The drawbacks of these passive markers are that their detection is sensitive to light conditions and that they must be of considerable size for detection and localization from a distance, causing issues for aerodynamics and carrying capacity.

These drawbacks can be addressed by using active markers instead. Works, such as [19], use LED markers in the visible range to localize robots. However, due to the copious intensity of visible light in general environments, especially due to the presence of sunlight, these markers are only suitable for use at short distances or in dimly lit environments, devoid of bright objects of colors similar to the markers. Infrared (IR) markers have also been used in the past [20], though they suffer from the same drawbacks outlined above. One notable approach for VRL of UAVs uses event-based cameras [21]. It shows promise due to the great speed these devices provide, but is currently limited by the high costs and low specifications of such cameras, as well as by the computational challenges related to processing their output in real time. Additionally, due to the presence of a moving background, mounting these devices on moving observers leads to a significant increase in information to be processed and to increased difficulty of background separation.

UAVs without markers can be detected by exploiting computer vision systems based on Machine Learning (ML) [22], [23]. This sensing modality is increasingly popular, due to its flexibility without the need for designing targets with the vision system in mind. However, as is outlined in one of the attached papers, their performance is dependent on the dataset used for their training, which limits their utility w.r.t. targets and environments significantly different from those appearing in the given dataset. Additionally, using real-time ML-based vision on a UAV requires specialized computational resources such as graphics processors being carried onboard.

The goals of this thesis and my work can be summarized as follows: first, to research the possibility of developing a system for mutual relative localization of UAVs that does not suffer from the above drawbacks; second, to design behaviors of UAVs that stabilizes them w.r.t. a target observed by VRL, such that the sensory noise, delays, perturbations, and other detrimental effects are accounted for; and, lastly, to apply VRL to cooperative multi-UAV flight, thus promoting the application of cooperating UAVs in real-world tasks where they show great promise.

The primary outcome of this research is the design and continuous development of a VRL system called UltraViolet Direction And Ranging ($UVDAR$), which the Multi-robot Systems Group (MRS) group uses onboard their UAVs. This system exploits the properties of solar radiation by using ultraviolet (UV) light to relatively localize UAVs and to enable direct communication between them in various challenging conditions, with special emphasis on bright exterior illumination with a minimal need for visual background separation. The

communication - and by extension identification - capabilities of the system use the active nature of the UV LED markers to modulate their light with binary signals that can be tracked in a camera image and decoded. This task requires simultaneous image tracking and extraction of signals encoded in bright spots that regularly disappear from view for a number of frames. The approach to this problem has seen numerous changes during its development and new modifications are still being researched by me, as well as my colleagues and students.

Using VRL in the control loop of a flying autonomous UAV presents numerous challenges that are rarely addressed in theoretical research. In my participation with the MRS Group at the Mohamed Bin Zayed International Robotics Challenge (MBZIRC) 2020 competition, I tackled such challenges and developed methods for autonomous aerial fire extinguishing with a partially actuated UAVs. These methods used onboard computer vision in a feedback loop with the extinguishing action. Such a system presents numerous challenges stemming from the detrimental effects that the motion of a UAV has on the image produced by an onboard camera. In addition, since fire extinguishing requires interaction with a target over time - such as continuously spraying an extinguishing liquid at a target - a specialized stabilization method was needed. The methods I developed enable active control counteracting the drifting of a UAV w.r.t. a target observed with a VRL, while also minimizing tilting of the UAV, which throws off the aim of the camera and the extinguishant stream.

Similar challenges occur when the targets of the VRL in question are mobile themselves, as is the case with UAVs cooperating based on a mutual VRL system. The creation of the *UVDAR* system enabled our group to implement numerous previously theoretical cooperative flight algorithms, including various swarms [24]–[27] and formation flights [28], [29]. Doing so revealed numerous challenges related to the properties of real-world environments and sensors, as well as some unforeseen opportunities related to the properties of the system.

To demonstrate the practicality of using *UVDAR* onboard autonomous UAVs as a source of relative localization in the control loop, we have first deployed a team of UAVs in a *swarm* following the Boids model. We were among the first [30] to deploy a UAV swarm without using external localization or explicit communication, basing the relative distance estimates only on the VRL system itself. These experiments have shown that under the simplistic, yet robust rules governing a swarm, *UVDAR* was a sufficiently reliable system for the group to maintain its compactness.

Taking note of the explosive growth in popularity of ML-based computer vision, we have leveraged *UVDAR* to generate an automatically annotated training dataset for these algorithms, thus addressing a key challenge in applying the popular computer vision modality to multi-UAV autonomous flights. This dataset has been published for use by the multi-robot research community.

A significant source of challenges lies in the fact that a real-world VRL output is not perfectly precise, but contains noise and even potential biases. The position of an object in an image corresponds to the relative direction towards said object, i.e., the relative *bearing*, which tends to have relatively good precision. *UVDAR*, as well as other VRL systems including ML-based vision software, can also provide an estimate of the distance to a target, albeit with significantly reduced precision. Notably, *UVDAR* also has the unique capability of providing an additional rough estimate of the relative orientation of a neighboring UAV. I have first demonstrated the advantage of this capability by performing a leader-follower flight of UAVs, designed such that the follower used the relative orientation estimate to align itself with a

specific side of the leader. This research marked the transition from swarms - a cooperative flight modality that we define by the interchangeability of agents following simplistic rules - to *formation flight*, where UAVs are made to assume a specific, defined formation in space.

Formation flight is the core topic of the final phases of my study, as the theoretical formation-driving algorithms in previous literature [31], [32] invariably presumed precise relative localization with insignificant statistical noise. Since such a level of precision is not available with real-world relative localization systems unless very specific conditions are enforced, they behave erratically and may even fail completely if the sensory noise is too large. The statistical distribution of the noise present in such measurements can be estimated and, as has been shown in the latest paper included in this thesis, such knowledge can be exploited. Specifically, with the technique proposed therein, it is possible for theoretical formation-driving algorithms to interface and successfully operate with real-world relative localization systems.

Chapter 2

Ultraviolet relative localization system *UVDAR*

The first part of my research concerns the development of a VRL system suitable for real-world cooperation between UAVs. As outlined in the introduction, various prior approaches exist. However, none have fulfilled the requirements we had for such a system, namely that it is independent of external infrastructure, does not require radio communication, and is independent of external lighting conditions within the range of exterior sunlight and total darkness. The main outcome of this research is a system called *UVDAR*, which is used onboard our UAVs. This system was developed in multiple phases and work on it is still ongoing as better camera sensors and processing is continuously becoming available, as well as by the requirements of new practical use-cases.

The system is based on the optical transmission of near-UV light from onboard LED markers, received by onboard UV-sensitive cameras. The advantage of this approach stems from the properties of outdoor sunlight. As we wanted our system to be operational in total darkness, the markers needed to be active light sources themselves. However, emitters in the visible spectrum, as well as in the near-IR range of the electromagnetic spectrum, were not deemed suitable as sunlight strongly emits within these ranges. Using such emitters runs an increased risk of unreliable detection due to the image background being too saturated by the refracted and reflected sunlight. In comparison, UV radiation of sunlight is considerably weaker, even fairly near to the visible range where common camera sensors like those we use (see Figure 2.1) are still sensitive. This means that, if we use suitable band-pass filters, the background in the camera image of this range is generally significantly darker than the active UV emitters, even during peak daylight. Examples of such camera images are shown in the included papers and in Figure 2.2. Note that the markers appear as small bright dots that are exceedingly easy to extract with image processing. The specific design of the markers and camera sensors is discussed in the attached paper

V. Walter, M. Saska, and A. Franchi, "Fast mutual relative localization of uavs using ultraviolet led markers," in *2018 International Conference on Unmanned Aircraft System (ICUAS 2018)*, © 2018 IEEE. Reprinted, with permission., 2018.

The next challenge to be addressed related to how the active markers are individually indistinguishable due to their shared appearance as bright dots or small blobs in the image. As a result, the individual UAVs carrying the markers are all anonymous in a single camera frame and can additionally be confused with specific types of glare and sharp specular reflections of sunlight. To address this challenge, I have developed a method of extracting a blinking identification signal from a short sequence of consecutive camera frames. The camera used allows for a limited frame-rate, typically set to 60 Hz. Thus, within the image sequence needed for signal extraction, the projection of a marker moves across the image. This image motion is the combined result of the movement of the target and of the observer carrying the camera. Note that, due to the blinking, the marker periodically disappears from view as well, and the combination of the image motion and blinking makes the markers susceptible to being incorrectly associated with the wrong previously seen marker. Therefore, a need existed for a method of extracting both the signal and image motion of the target at the same time.

Figure 2.1: Camera and lens equipped with two different ultraviolet filters used in our early tests of ultraviolet vision are shown in the top row. This configuration proved the concept, but was deemed unsuitable for deployment onboard UAV due to size, costs, and incompatibility with fisheye lenses. The bottom row shows the hardware components of our most commonly used and current *UVDAR* sensor implementation.

Figure 2.2: View from a UV camera used in the *UVDAR* system. The three bright dots are the active markers attached to a neighboring UAV. This view is captured in a desert environment at noon, showing the high level of robustness to exterior illumination.

The solution I developed is to use the Hough Transform (HT) to extract a 3D line that approximates the motion of the markers - in the space of X-Y-time - from the detections accumulated by a short sample of previous camera frames. Using this method, it is possible to extract both the motion of the marker in the image, as well as its blinking signal by reading out frames where the marker was detected along the extracted trajectory. The details of this signal extraction method are provided in the attached paper

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 *14th IEEE International Conference on Automation Science and Engineering (CASE 2018)*, © 2018 IEEE. Reprinted, with permission., 2018.

The initial implementation of the above system used crude signals in the form of square waves with a 0.5 duty cycle, differentiated by their frequencies. This proved to be effective to differentiate a number of markers, as well as to better suppress parasitic detections of sunlight based on them not matching the expected pattern. However, the number of such signals was limited by the maximal frequency stemming from the Nyquist limit of the cameras used, as well as by the minimal frequency defined by the time during which the image motion of the markers can be reasonably expected to be sufficiently straight for a linear approximation. Additionally, the consecutive frequencies of the signal set needed to have periods separated by, at the bare minimum, one camera frame. In practice with the software and hardware setup at our disposal, the above signaling allowed for five individual signal IDs at most, limiting the scalability of a multi-UAV group using the system. This limitation was overcome by changing the signaling used to a more complex one, where we generated a set of periodic binary signals of a set length in terms of camera frames. These signals needed to conform to a set of constraints, such as having specific minimum circular hamming distances between each pair, as well as a set of other empirical constraints. The design of this new signaling method is shown in the included paper

D. B. Licea, V. Walter, M. Ghogho, and M. Saska, "Optical communication-based identification for multi-uav systems: Theory and practice," 2023. arXiv: `2302.04770` `[eess.SP]` Under review in Autonomous Robots.

This step upgraded the set of usable identities to over twenty without changing the size of the sample set or camera frame-rate. It allowed our group to perform large-scale swarm flights [27], as well as formation flights where multiple signals are used on each individual UAV to extract relative orientations [28], [29]. An example of a camera view with annotated markers attached to two neighbor UAVs is shown in Figure 2.3.

Figure 2.3: View from a UV camera used in the *UVDAR* system from a multi-UAV scenario. The markers attached to two neighboring UAVs are automatically annotated with their respective IDs based on their transmitted blinking signal.

Notably, besides extensive internal use by our our group, the system was exploited in cooperation with researchers from the Technology Innovation Institute (TII) institute [33]–[37], and it was even used by researchers independently of us [38]. It was also used in education, both by six of our bachelor and master students within their thesis projects, as well as by over a hundred external students as part of the IEEE RAS Summer School on Multi-Robot Systems[1] organized by our group.

The software used in *UVDAR* for processing camera stream is available on-line as an open-source Robot Operating System (ROS) package, found at `https://github.com/ctu-mrs/uvdar_core`.

---

[1]`https://mrs.felk.cvut.cz/summer-school-2020/`

Fast Mutual Relative Localization of UAVs using Ultraviolet LED Markers

Viktor Walter[1], Martin Saska[1], Antonio Franchi[2]

Abstract— This paper proposes a new methodology for outdoor mutual relative localization of UAVs equipped with active ultraviolet markers and a suitable camera with specialized bandpass filters. Mutual relative localization is a crucial tool for formation preservation, swarming and cooperative task completion in scenarios in which UAVs share working space in small relative distances. In most current systems of compact UAV swarms the localization of particular UAVs is based on the data obtained from motion capture systems for indoor experiments or on precise RTK-GNSS data outdoor. Such an external infrastructure is unavailable in most of real multi-UAV applications and often cannot be pre-installed. To account for such situations, as well as to make the system more autonomous, reliance on onboard sensors only is desirable. In the proposed approach, we rely on ultraviolet LED markers, that emit light in frequencies that are less common in nature than the visible light or infrared radiation, especially in high intensities. Additionally, common camera sensors are sensitive to ultraviolet light, making the addition of a filter the only necessary modification, keeping the platform low-cost, which is one of the key requirements on swarm systems. This also allows for a smaller size of the markers to be sufficient, without burdening the processing resources. Thus the proposed system aspires to be an enabling technology for deployment of large swarms of possibly micro-scale aerial vehicles in real world conditions and without any dependency on an external infrastructure.

I. INTRODUCTION

Mutual relative localization of flying robots is indispensable in many real-world applications that require deployment of multiple Unmanned Aerial Vehicles (UAVs) sharing the same workspace in small relative mutual distances. Using compact multi-UAV systems brings numerous benefits including cooperative task completion, extension of reach of a single robot and distribution of capabilities into independent members. Moreover, several tasks that are not solvable by a single robot do exist and some of them were successfully solved by teams of UAVs developed by the Multi-robot Systems group at CTU in Prague [3] by employing onboard visual mutual relative localization - see Fig. 1 a) and c) for examples. In this paper, we propose a novel robust method of infrastructure-independent relative localization for flights of multiple UAVs, applicable for outdoor environments, as
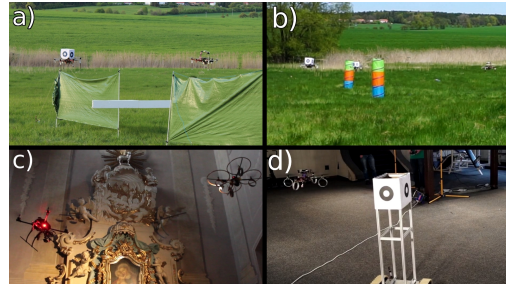

Fig. 1: Examples of applications of mutual relative localization developed by our group. a) Cooperative carrying of large objects, b) simultaneous flight through forest-like environments using bio-inspired swarming rules, c) mapping of historical buildings by a formation of camera-carrying UAV and spotlight-carrying UAVs to obtain footage with fixed illumination angle d) heterogeneous formation of self-stabilised UAV-UGV.

well as the indoors. The method is based on the application of markers composed of ultraviolet LEDs on the UAVs, in addition to equipping the observer UAVs with cameras with fisheye lenses and specialized bandpass filters. The relative pose of the observed UAVs can then be retrieved easily from the image, where the markers are visible as bright spots on dark background that can be located with little processing. Our intention is to provide each swarm member with as complete as possible information on the state of all UAVs in close proximity without any the of communication among the robots or with a base station. Fulfilling this goal requires direct sensing of mutual relative distances, information about the relative bearings and, if possible, also about their relative headings.

As can be seen in numerous examples of flocking of animals, such information can be effectively obtained through vision [1]. In our previous works on mutual localization [2], [3], passive markers in conjunction with an object detection algorithm have been used to achieve the same sensory ability. This vision-based system has been deployed in numerous indoor and outdoor experiments [4], [5], [6], where multiple drawbacks, such as strong lighting conditions dependency, large size of the markers, limited operational space and computational complexity have been identified. The most significant is the fact that the changing outdoor lighting conditions may prevent consistent recognition of the markers. The system proposed in this paper aims to alleviate these drawbacks and to further enlarge the application domain of UAV swarms.
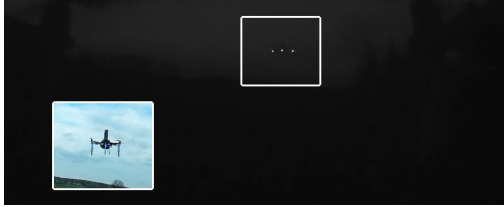
Fig. 2: Example of an image from the onboard camera with filter of a hexarotor UAV equipped with 6 markers on its arms. The picture was taken at noon, with the exposure rate set to 1000 μs. Lower left shows the same view from a conventional camera.

Fig. 3: An extreme case of long range and hard to separate background. In the visible spectrum the UAV is difficult to locate even by the human eye and today's popular CNN methods [7], while in UV the three markers are clearly visible as unique peaks in brightness and can easily be detected by the proposed system.

A. State of the art and contributions

Most of the multi-UAV experiments that require localization have been realized in laboratory conditions relying on an absolute measurement by a motion capture system (Optitrack[4], VICON[5], etc.) [8] and the relative pose measurement is emulated by calculating it from a source of absolute pose as a placeholder [9]. Outdoor cooperative flights [10], [11] tend to rely on GNSS (Global Navigation Satellite System) if a close proximity of robots is not required or on RTK-GNSS data (provided in our case by Tarsus BX305[6]), which may achieve the precision of ±10 mm. Obviously, these approaches suffer from the necessity to pre-install an external infrastructure (motion capture cameras or RTK base-station) which precludes flights in environments that are unknown or difficult to reach by the operators themselves. Another problem is transition from outdoor to indoor environments, flight near elevated objects of larger volume - buildings, rock formations, etc. - or flight in an unknown, cluttered or inaccessible enclosed environment. Additionally, continual reception of such external information requires wireless communication, which is subject to limited range and interference both from unrelated sources and by the units themselves in the case of a large swarm. This makes the

system difficult to scale up to larger number of units, which is the main idea of robotic swarms. Another challenge caused by the infrastructure dependence the task may easily be deliberately obstructed by interfering with the infrastructure, such as by introduction of artificial radio interference on key radio frequencies. In order to be reliable even in such circumstances, the units need to be able to independently avoid damage and complete their mission. If the UAVs are flying in a formation, they should be able to preserve it or keep their mutual distances within safe ranges, which is reliably enabled by the proposed approach.

Numerous principles of direct mutual localization can be found in literature. From a theoretical point of view the mutual localization problem in group of robots boils down to the (bearing) rigidity problem, see [12] and reference therein for an introduction to this concept. The multi-agent mutual localization problem has been also faced in the case in which measurements do not provide the identity of the measured robot, i.e., they are anonymous [9].

Relative localization of noise-emitting objects such as UAVs was successfully tested in [13]. This method, however, requires a large and highly specialized equipment, while providing only an approximate relative bearing of the target and being sensitive to acoustic environmental noise.

Another approach was used in [14], where point-clouds obtained from RGB-D cameras attached to each unit are aligned to obtain their relative poses. These sensors have severely limited range and field of view and the algorithm is computationally too complex for most onboard computers of lightweight UAVs. Similarly, [15] used the alignment of lines detected in images of the environment of the UAVs to establish their relative positions. Such an approach can be applied efficiently in an environment with multiple straight lines, such as in streets or offices, but not in natural environments where straight lines are rare. Both of the aforementioned approaches additionally require communication between the units, at fairly high bandwidths.

Numerous experimental solutions based on vision and mutual observation of UAV and UGVs equipped with known geometrical markers were tested [16], [17]. Our previous solution uses circular visual markers [2], [3] for mutual localization in small swarms of UAVs [5], [6] and in heterogeneous formations [4]. The main disadvantage of these methods is the sensitivity to light conditions, computational complexity and the physical dimensions of these markers. The large size of the markers (see fig. 1-a), needed for proper detection from reasonable distances and angle range leads to problems connected to aerodynamics and maintenance difficulties. In addition, these markers are highly susceptible to a partial occlusion that can prevent detection.

A basic and often used approach is to apply a simple, easy to segment color-based markers that work well in controlled light conditions of laboratory environments [18], [19]. While in a laboratory it is easy to apply active or passive markers of a color that we ensure is not otherwise present in the environment, this is seemingly unfeasible in outdoor conditions, since in nature as well as in urban scenarios, all

---

[4]`http://optitrack.com/`
[5]`https://www.vicon.com/motion-capture/engineering`
[6]`https://cdn.shopify.com/s/files/1/0928/6900/files/\\Datasheet\_BX305\_Kit\_433\_915\_EN\_0913.pdf`

2.1. ARTICLE 1: ULTRAVIOLET VISION FOR RELATIVE LOCALIZATION OF UAVs

colors of the visible spectrum are naturally present.

The closest approach to the proposed system can be found in [20], where UAV localization through infrared lights blinking in the order of kilohertz attached to helicopters is implemented. Such frequency allows for the markers tracked even with aggressive flight maneuvers. This was achieved by using an event-based camera. The solution in [20] was only tested in indoor environments, with the camera being statically placed in the room, instead of onboard of a UAV. The use of a heavy and expensive specialized camera makes it unsuitable for outdoor onboard deployment. Additionally, the low resolution of contemporary event-based cameras decrease the precision and the range of detection.

The system presented here is based on the observation that while the colors of the visible spectrum, as well as a wide band of the infrared wavelengths, are present with relatively high intensity in normal sunlight, the ultraviolet light is significantly less intense. This means that in the UV spectrum the natural environments are dark and when an object bright in the ultraviolet is observed, it is likely to be artificial. Exploiting this fact does not even require a new specialized sensor, since common monochromatic digital cameras tend to be receptive to near ultraviolet frequencies, and need only be modified with a proper band-pass filter. The UV light can then be used as a robust and easy detect feature. The only false positives in detection being the sun itself and its specular reflections, while even most of artificial light sources emit little to none UV radiation.

To sum up the contribution of the paper related to the current literature, we emphasize that the proposed solution combines knowledge gained during hundreds of flights with multiple closely cooperating UAVs in realistic indoor and outdoor conditions using different state-of-the-art localization systems and a theoretical analyses of sources of disturbance that lead to false positive detections in common workspaces of UAV systems. Based on this data we propose the HW design of a system made of multiple onboard UV light sources that reliably provides the required information (distance and bearing of neighbouring vehicles) in all possible configurations of the team. In comparison with state-of-the-art solutions, our solution presents better reliability w.r.t different weather conditions and precision, while significantly reducing size and weight of the overall system and the required computational power. We provide two approaches for mutual localization using this HW setup. The first minimalistic approach uses a single camera and a single UV LED on each UAV to provide bearing information and a rough estimation of the relative distance. This method together with the work [21], where we have shown that such a sensory information (even anonymous) is sufficient for reliable coherent swarming, provides a complex solution for deployment of large swarms of micro aerial vehicles. The scalability is shown in [21] by a surprising observation that the swarm stability and coherence increases with number of swarm members even with such a limited sensory input. The second approach presented in this paper exploits possibility of using multiple LEDs onboard of UAVs to increase pre-
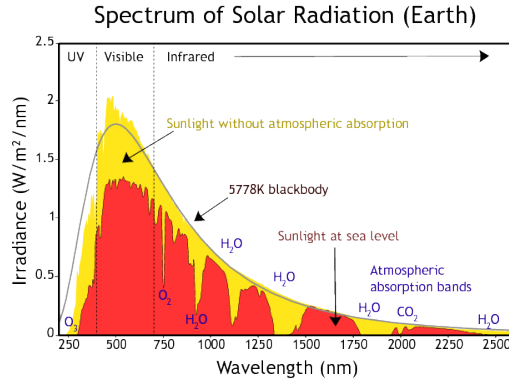

Fig. 4: The spectrum of solar radiation in wavelengths near the visible spectrum. Notice the rapid decrease in irradiance in the UV region.[7]

cision of the distance measurement and operational space. In fact, this approach exploits full size of the UAVs, putting the LEDs as far apart as possible, which increases baseline used for the distance estimation, resulting in higher precision, in comparison with passive markers that are always significantly smaller that the UAV itself.

The paper is structured as follows. Section II deals with the theoretical background applied in the design of the system. Section III comprises the overview of the hardware used. Section IV explains the methodology used in estimating the mutual location of the target (neighbouring UAV) as well as the system identification. Finally, section V summarizes the results of experimental testing of the system.

II. THEORETICAL BACKGROUND

A. The ultraviolet markers and sensors

Using UV light for mutual localization is an obvious approach for deployment of swarms in real outdoor environments, as it was the case in our experiments, but not in typical laboratory experiments [8], [22]. The solar radiation approximates the blackbody radiation, with peak intensity centered on the visible light (see figure 4). This means that the shape of the intensity to wavelength characteristic is asymmetric w.r.t. the wavelength. The intensity decays considerably slower with growing wavelengths than it does with diminishing wavelength. Ultraviolet parts of sunlight are therefore significantly less intense than the visible light and the infrared, even relatively close to the visible spectrum. We can exploit this observation to apply active markers that can be easily distinguished from the multicolored natural backgrounds purely based on the intensity. In the implementation, this requires physical band-pass filtering tuned to the ultraviolet wavelengths of the markers.

[7]https://commons.wikimedia.org/w/index.php?title=
File:Solar_spectrum_ita.svg&oldid=261911890

A gray-scale camera is more suitable for this task since it is less selective w.r.t. the wavelength than RGB cameras and is thus more sensitive to light outside of the visible spectrum. Note, that while the sun does still emit considerable amounts of UV radiation, this radiation is spread across a range of wavelengths and most of it is filtered out by the atmosphere. Additionally, it does not reflect well on solid natural surfaces. With low enough exposure rate of the camera, the only objects that will locally saturate the resulting image to white will be the markers, directly observed sun and some of its specular reflections. The appropriate exposure rate depends on the required maximal detection distance. Our tests have shown that the UV light refracted through the atmosphere and reflected from matte surfaces is normally too dim to prevent the detection. The image of the sun and its reflections should be a minor issue which can be accounted for by using knowledge of the size or other specific characteristics of the spots caused by the sun or the knowledge of sun position based on current place and time. The possibility to set the exposure so that the markers will create saturated white spots on dark background can be used to binarize the image using simple static thresholding, as opposed to the more computationally demanding adaptive thresholding needed for arbitrary lighting conditions. The image spots caused by artificial light sources will be saturated regardless of the time of the day.

In order for this system to be suitable for mutual localization of multiple UAVs, the camera has to be able to observe the surroundings in a large field of view. This can be achieved by using a fisheye lens. A potential drawback that has to be addressed is the high rate of decay of UV light in common types of glass, proportional to the specific frequency. In our tests, this has limited the wavelength of the UV light that we could use. Light sources radiating at 365 nm were suppressed to the extent that they were mostly invisible to the camera, while light sources with wavelengths of 395 nm were clearly visible. Specialized lenses permeable to high-frequency UV that also have wide FoV and are portable could not be found on the market at the current time. Moreover, the infrared filter applied to the lenses by the manufacturer also blocks out UV light. A shorter wavelength would allow for better filtering of of the sunlight. Despite this, the 395 nm light sources and filters proved to be sufficient in our experiments.

III. Hardware overview

The camera used in our experiments is based on the *Matrix Vision mvBlueFOX-MLC200wG* sensor, equipped with the *Sunnex DSL215* fisheye lens and *Midopt BP365-R6* ultraviolet bandpass filter.

The *mvBlueFOX-MLC200wG* (figure 5-c) is a greyscale CMOS sensor with a resolution of 752×480 pixels, maximum frame-rate of 93 without binning and quantum efficiency at 395 nm of ≈ 34 %[8] In our experiments we were able to achieve a maximum frame-rate of 70 Hz with the exposure rate set to 1000 µs. The *DSL215* is a fisheye lens

[8]`https://www.matrix-vision.com/\\USB2.`
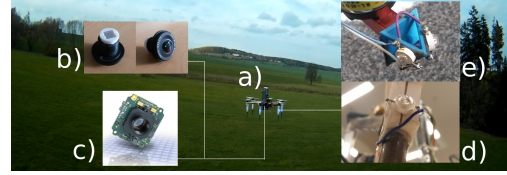`0-single-board-camera-mvbluefox-mlc.html`

Fig. 5: Summary of the proposed system components. The UAV (a) is equipped by a *mvBlueFOX-MLC200wG* camera sensor (c) with *DSL215* lens with *BP365-R6* bandpass filter that allows it to observe and localize ultraviolet LED-based markers (d) or (e)

with the maximal FoV of 185°. This value, however only applies to the horizontal field of view with the *MLC200wG* sensor. [9] The *BP365-R6* is a miniature interference-based optical bandpass filter for ultraviolet imaging[10]. The custom size of $6 \times 6 \times 1$ mm allowed us to attach it between the lens and the sensor so that the whole image is covered, as seen in figure 5-b. For the markers, we have selected the *ProLight PM2B-1LLE*, (figure 5-d) which is a high power ultraviolet LED, with the maximum of emission centered on the wavelength of 395 nm and with Lambertian radiation pattern. [11]

This beacon-sensor system is small, lightweight and relatively affordable, and thus ideal for deployment with small UAVs.

IV. Method description

A. System identification

1) Camera calibration: In order to convert the image positions of the detected spots corresponding to the markers on the UAVs into relative bearing vectors, it was necessary to perform geometric calibration of the camera. We have selected the model described in [23], suitable for cameras with wide FoV, that translates pixel positions directly into bearing vectors.

The parameters of the camera projection are affected by the different index of refraction of UV light compared to the visible light , as well as by small eccentricities of the lens mount. To account for these factors we calibrated the camera with the band-pass filter on and the lens attached in the final position. For the chessboard-type calibration pattern to be fully visible, the pattern had to be illuminated by a UV light source, with the exposure rate and threshold manually adjusted for the different angles of view, some parts of the pattern became overbright or overly dark depending on the angle of reflection. After obtaining the images, the calibration was done semi-automatically using the *OCamCalib* toolbox [12]. The toolbox itself provides a method of converting pixel position into a unit bearing vectors of the markers, here denoted $c2w$ and the reverse function $w2c$.

[9]`http://www.optics-online.com/OOL/dsl/dsl215.pdf`
[10]`http://midopt.com/filters/bp365/`
[11]`http://datasheet.octopart.com/\\`
`PM2B-1LLE-Prolight-datasheet-41916849.pdf`
[12]`https://sites.google.com/site/scarabotix/`
`ocamcalib-toolbox`, [24]

2.1. ARTICLE 1: ULTRAVIOLET VISION FOR RELATIVE LOCALIZATION OF UAVs

The angle ϕ between the optical axis of the camera and an optical ray corresponding to a point in the image was modeled as a linear function of the distance r of the point from the optical center in pixels. The slope of the function was measured to be

$$k = \phi/r = 3.7673 \times 10^{-3} \text{ rad px}^{-1}. \tag{1}$$

2) Spot size: An interesting imaging effect that can be exploited in this setup is the blooming effect of the camera. The markers used here are small LEDs, which with an idealized camera each LED would be projected into a single point on the sensor resulting in a single bright pixel. With real-world cameras, these markers are shown in the image as spots of sizes depending on the distance. This effect occurs due to a combination of monochromatic optical aberrations of the lens and of the finite capacity of the CMOS elements in the sensor causing excess charge being spilled over to the surrounding pixels. The exact analysis of the nature and relative impacts of these phenomena is beyond the scope of this paper.

The sizes of the spots can, however, be analyzed w.r.t. the distance of the marker (see figure 6) and thus used to give a rough estimate of the distance of the marker. Since the geometry of these spots depends on many unknown variables, and due to the finite resolution of the camera, the true position of the ray incidence within such a spot is ambiguous. The spots shrink down to the size of a single pixel at a certain distance from the camera, depending on the resolution, exposure rate, type of sensor and the radiation intensity of the marker in the direction of the camera. In order to preserve high output rate and ease of processing, we store the sizes of these spots in terms the number of pixels S after thresholding. To represent the position of the spot, we store the x and y image coordinates of the middle pixel in order of contour filling.

Since the tend to be slightly blurry around their edges, the size of the spots after binarization depends on the selected threshold.

While the estimation of distance based on such an imprecise information is not ideal, it can inform of a neighbor breaching certain safety radii or leaving the minimal mutual distance. With the algorithm presented in [21] and the proposed approach, we can achieve coherent swarming using a minimal required mutual localization setup consisting of one simple camera and LED per unit (total mass of 50g), which opens new perspectives in use of micro aerial vehicles (MAVs).

Additionally, knowing the characteristics of the spot sizes $S(l)$ w.r.t. the distance from the camera l is useful for establishing margins of error for measurements based on the estimated bearing vectors. For this purpose, we can select a function $S_{max}(l)$ such that the values will lay under it. These were measured in experiments and can be seen in the next sections.

B. Directional vector estimation

Measuring the relative bearing between UAVs is one of the main requirements to let them stably fly in swarms with

Fig. 6: Relative difference in sizes of spots generated by markers attached to quadrotor UAVs approximately 2 m (lower triplet) and 5 m (upper triplet) away from the camera.

a predefined shape. However, it is not always necessary for each UAV to measure all the relative bearings w.r.t. all the other members. Indeed, Franchi et al. [22], [9] have shown that a stable, controllable swarm with a predetermined shape can be achieved by resorting only to a certain minimum number of directed observations of relative bearings between the swarm members with known identities. It was also demonstrated [25] that it is possible to reconstruct the shape of the swarm purely based on relative bearings of the unit even when the identities of the observed neighbors are unknown. With our system, such control algorithm will be applicable not only to units with limited processing and data transfer capabilities but also to outdoor applications. Since we have already calibrated the camera, we can convert a pixel position m_i to the relative bearing vector v_i towards a single marker as

$$v_i = c2w(m_i). \tag{2}$$

In reality the pixel position m_i corresponds to a range of bearing vectors that are projected to the same CMOS element, while the vector v_i corresponds to the center of that range. In the worst case, the spot is a linear chain and the true bearing vector corresponds to the farthest point of the pixel located the farthest away from the pixel we have stored to represent the spot position. The calculated relative bearing vector v_i can, therefore, differ from the true bearing vector v_b at most by the angle

$$\epsilon = k \left(S_{max}(l) - \frac{1}{2} \right). \tag{3}$$

C. Distance estimation based on image distance of two markers

The distance of an object in an image can be estimated based on the distance of two points M_1 and M_2 on the object (two LEDs of known relative position onboard of the UAV) and the angle α they form with the camera origin C. This requires for the camera to be calibrated, and is only applicable if we have a reasonable assumption on the angle between $\overleftrightarrow{M_1 M_2}$ and the direction towards the camera. For this method to be used for UAV localization, there needs to be at least a pair of visible markers. These two markers should not deviate too much from perpendicular alignment w.r.t. the direction towards the camera.

In order to limit the dependence of the precision of the distance estimation on the relative orientation of the observed UAV, the markers should be placed along a circle centered on the UAV local frame of reference, with equal distances between adjacent markers. Additionally, these markers should be spaced as far apart as possible, while still allowing
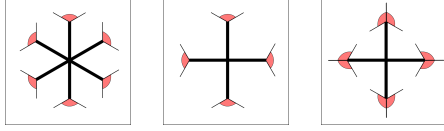
13

Fig. 7: Allignments of LED markers for a hexarotor, and a quadrotor. In the center layout, only a single marker would be visible from some directions, as opposed to the rightmost layout, with markers consisting of two LEDs at the mutual angle of $120°$.
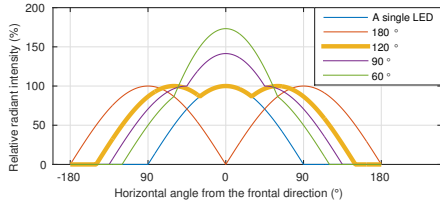

Fig. 8: The relative radiant intensities w.r.t. the direction for two combined, symmetrically aligned ideal Lambertian radiators. With the mutual angle being $120°$ the intensity in the central direction is the same as for a single radiator aimed in that direction.

Fig. 9: Schematic of the two borderline algnments of a hexarotor and the camera. With the camera in position C_A, the observed pair is in perpendicular alignment w.r.t. the camera. In position C_B the points are in the $30°$ alignment w.r.t. the camera.

at least a single adjacent pair to be visible from every direcion. For multirotor UAVs with a star-like layout, with the propellers attached to equally spaced arms extending from the center, the best choice is to attach these markers at the ends of these arms. In other cases, they can be attached to a protective cage or on specific additional components, to ensure radial symmetry. For a hexarotor UAV with equal arm lengths and internal angles, which is our most common use-case, the markers, composed of a single LED each, are attached to the ends of the arms. In this case a single LED per arm is sufficient, since the intensity of radiation in a Lambertian radiator (such as our chosen LEDs) at the angle of $60°$ will not drop below 50 % of the intensity in the frontal direction. In a similar quadrotor UAV, the markers must be composed of symmetrically angled pair of LEDs to account for the negligible intensities of radiation in the perpendicular direction in a single LED (see figure 7 and 5-e).

We recommend for the angle between these two LEDs to be $120°$ in order to get approximately the same radiation intensity in the direction away from the center as with a single centered LED (see figure 8).

For a hexarotor with such markers we can estimate the distance in a range defined by two borderline assumptions:

1) The observed pair is in perpendicular alignment w.r.t. the direction towards the camera - camera in position C_A in figure 9;

2) One marker of the pair is on the connecting line of the hexagonal frame - camera in position C_B in figure 9.

Which situation is currently closer to being the case unknown. While hexarotor is used as an example, similar borderline assumptions can be defined for quadrotors, octarotors, etc.
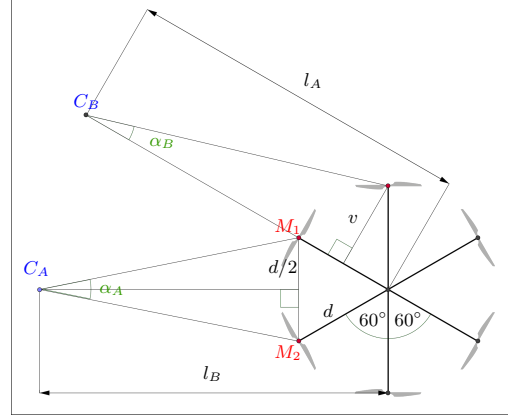
Depending on whether we need to check for UAVs being

too close or too far away, we can select either calculation. In the case 1) the UAV is more likely to be closer than estimated, while in the case 2) it is more likely to be further away. In a swarming algorithm such as [21] the UAVs need to compare the distances of neighbors with two margins, the far limit and the near limit, between which the distance should be kept. In order to make such swarming more reliable, it makes sense to calculate with case 1) for the far limit and with case 2) for the near limit. This is also applicable for tasks such as cooperative carrying of large objects (see figure 1-a), where these limits are connected with the level of control over the object, collision avoidance and energy conservation.

The distance can be calculated from the triangle formed by the camera C and the two markers M_1 and M_2. The parameter d is distance between M_1 and M_2 and α is $\measuredangle M_1 C M_2$. The relative bearing vectors from the camera towards points M_1 and M_2 are denoted $\mathbf{v_1}$ and $\mathbf{v_2}$.

$$\alpha = arccos\left(\mathbf{v_1} \cdot \mathbf{v_2}\right) = arccos\left(c2w(\mathbf{m_1}) \cdot c2w\left(\mathbf{m_2}\right)\right) \quad (4)$$

In the case 1), the distance l_A to the UAV center can be expressed as

$$l_A = \left(\frac{d}{2}\right) cot\left(\frac{\alpha}{2}\right) + v, \quad (5)$$

while in case 2), the distance l_B can be expressed as

$$l_B = v\ cot\left(\alpha\right) + \left(\frac{d}{2}\right). \quad (6)$$

The symbol v here stands for the height of an equilateral triangle with side d.

The intensity of radiation in a Lambertian light source, such as the one we are using is roughly proportional to the cosine of the angle from the axis. It therefore is more common to see only two of the six markers than three or four.

2.1. ARTICLE 1: ULTRAVIOLET VISION FOR RELATIVE LOCALIZATION OF UAVS

To establish error margins for the two cases at ground truth distance l, we take the assumption that the wrong case was presumed, so that while calculating for case 1), the real situation corresponded to case 2) and vice-versa. If in the case 2) the calculation for the case 1) was used, the relation of the estimated distance to the real distance would be

$$l_{aerr} = \left(\frac{d}{2}\right) \; cot\left(arctan\left(\frac{v}{l-\frac{d}{2}}\right)/2\right) + v. \quad (7)$$

In the converse case, the erroneous selection of calculation results in the estimate

$$l_{berr} = v \; cot\left(2 \; arctan\left(\frac{\frac{d}{2}}{l-v}\right)\right) + \left(\frac{d}{2}\right). \quad (8)$$

These margins should be additionally expanded by the effects of finite resolution and spot size. Due to foreshortening, as the UAV retreats away from the camera, the change of mutual distance of the two markers in the image becomes less and less pronounced. This is more significant when the mutual distance of the images of the markers becomes comparable with the size of a pixel. As was the case in the directional vector estimation, the maximum angular error ϵ in the direction of the vector $\mathbf{v_1}$ is equal to $k(S_{max}(l) - \frac{1}{2})$

The final maximal value of distance estimation in the first case is then:

$$l_{amax} = \left(\frac{d}{2}\right) \; cot\left(\frac{arctan\left(\frac{v}{l-\frac{d}{2}}\right) - k \; (S_{max}(l) - \frac{1}{2})}{2}\right) + v \quad (9)$$

The minimal value for this case is:

$$l_{amin} = \left(\frac{d}{2}\right) \; cot\left(arctan\left(\frac{\frac{d}{2}}{l-v}\right) + k \; (S_{max}(l) - \frac{1}{2})\right) + v \quad (10)$$

In the other case, where we presume that one of the spots in the observed pair corresponds to a marker on the line connecting the camera and the center of the UAV, the maxima and minima of the estimation can be expressed similarly:

$$l_{bmax} = v \; cot\left(arctan\left(\frac{v}{l-\frac{d}{2}}\right) - k \; (S_{max}(l) - \frac{1}{2})\right) + \left(\frac{d}{2}\right) \quad (11)$$

$$l_{bmin} = v \; cot\left(2 \; arctan\left(\frac{\frac{d}{2}}{l-v}\right) + k \; (S_{max}(l) - \frac{1}{2})\right) + \left(\frac{d}{2}\right) \quad (12)$$

D. Full pose estimation

For a more precise position estimation of a neighbor UAV that is more suitable for 3D environments and returns orientation as well, we might consider using a Perspective-n-Point method. This may prove quite challenging in implementation, due to the anonymity of the observed points and due to the diminished visibility of the markers not facing the camera. Additionally, these methods are computationally more complex, which may reduce the output rate. One way to increase the number simultaneously of visible points is to increase the number of the LEDs on the UAV. This may be done in two possible ways:

1) by composing each marker out of multiple LEDs, as can be seen in figure 5-e
2) by adding more single-LED markers for denser coverage.

Care should be taken with such a modification, since if the distribution of the markers is too dense then they will tend merging in the image into a single spot. The problem of anonymity can also be addressed this way, by constructing patterns that can be matched with a known template without ambiguity.

Another potentially effective approach under consideration is to encode information such as individual ID of a marker into blinking patterns.

V. EXPERIMENTS

A. Distance estimation based on spot size

To evaluate the relationship between the mutual distance and the size of the bright spot an *Optitrack* motion-capture system was used to record the ground-truth distance between the camera and a single LED in its view, while the size of the spot was being recorded simultaneously. This procedure was repeated for multiple exposure rates. The binarization threshold used for segmentation of the bright spots was set to 200 out of 255, or 78.42% of the saturated brightness (for the characteristics, see figure 10). These tests seem to indicate, that the best exposure rate for detecting whether the neighbors are within a reasonable distance with this setup is 1000 ms, when the spot shrank consistently to the size of a single pixel at 6.62 m while still sufficiently filtering out the ambient UV illumination. Therefore, a very simple and low-cost approach (only one camera and one LED is required) can be used for an effective collision avoidance as it provides sufficient safety distance. For example the safety distance used in Multi-robot Systems group in [26] was 5 m. Within the range from 3.14 m to 6.62 m the occurence of spots with the size of 1 pixel become common. For distances smaller than 3.14 m the non-linear the characteristic can be used for a rough, but quantitative distance estimation in applications, where the UAVs have to operate within very close mutual distances. From the characteristic for the 1000 μs exposure rate (figure 11) we have derived the parameters of an approximating equation (13) in the form of a quadratic-hyperbolic function rounded to the nearest integer.

$$S_{max} = \lfloor 1.3398 + \frac{31.4704}{(x - 0.0154)^2} \rfloor. \quad (13)$$

A quadratic-hyperbolic function was selected because the number of saturated pixels from a single ray depends roughly on the overall energy that has been received by the sensor from the marker, which in turn is governed by the inverse square law.

A different exposure rate can be selected for tasks where closer or greater mutual distances are required (note the different single-pixel thresholds in figure 10). For example,
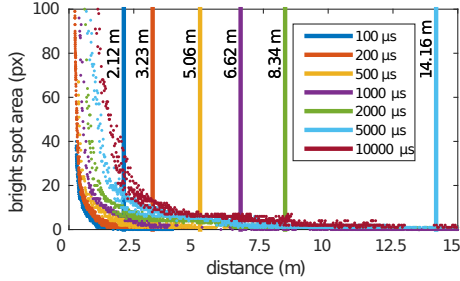
Fig. 10: Distance dependence of the bright spot size for multiple exposure rates. The vertical lines denote distances beyond which the spots are consistently reduced to a single pixel.
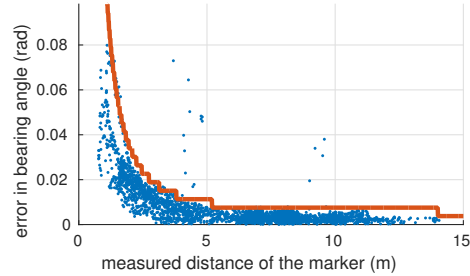

Fig. 12: Angles between the estimated bearing vectors based on the image and the true bearing vectors. The red line denotes the expected maximal errors based on equation 3
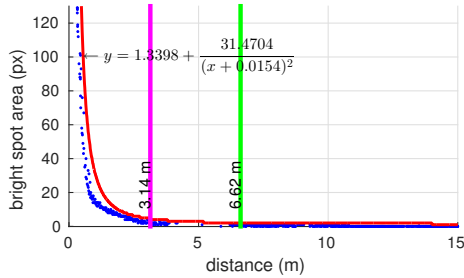

Fig. 11: The spot size to distance characteristic for the exposure rate of 1000 μs. The green line denotes the distance where the spot becomes a single pixel. The magenta line denotes the shortest distance, where spots with the size of 1 px appear. The red curve is the approximating function of the maximum possible spot size.

in applications where very tight formations of micro aerial vehicles are needed we can select short exposure rate to reach a higher precision in distance estimation at short distances, in addition to better selectivity w.r.t. other light sources.

If the margins used in a swarming algorithm are derived from the spot sizes, then we can expand or shrink a formation simply by changing the exposure rate. This can even be done dynamically based on the circumstances, altering the swarm parameters on-the-fly (for instance by changing the constants in a Boids model [27] we can change the swarm size and thus adjust the working area).

Using a high exposition rate presents a trade-off, since it allows for detection and quantitative distance estimation in larger mutual distances, but has the side-effect of lower selectivity w.r.t. other light sources and thus lower robustness.

B. Bearing vector estimation precision

In order to verify the precision of the relative bearing vector estimation and the distance estimation based on the mutual distance of two spots in an image, we have equipped a hexarotor frame with an arm length of 0.4 m with UV light sources on the end of each arm. Precision and relia-

bility of the mutual distance estimation were evaluated in different relative positions and orientations using *Optitrack* as a ground truth.

Lines connecting the camera with the potentially visible markers were calculated from the ground truth poses of the camera and the hexarotor frame. Relative bearing vectors were estimated from the bright spots in the image. The bearing vectors and connecting lines were compared and the angles between the closest found matches were stored. Figure 12 shows, that the predicted maximal error in the estimation according to equation (3) holds.

C. Neighbor distance from mutual distance of points

In the same dataset as used previously, the pair of adjacent spots with the greatest mutual distance in the image was used as the basis for the distance estimation. The distance of the frame center from the camera was calculated both according to equation 5 and equation 6. The estimated distances compared to the ground truth obtained from the *OptiTrack* system can be seen in figure 13, together with plots of the predicted margins of error based on the previously obtained function $S_{max}(l)$. Note, that in the first case most estimates are greater than the ground-truth, while in the other case the opposite is true.

Figure 14 shows the relative errors in estimations in both cases.

Within the tested range the error increases roughly linearly with the distance. This is conforms to the expectations and is of no concern in forming a swarm, since the precision is only low outside of collision range.

The precision is decreased by the randomized angle of the observed pair of markers so that the results would be consistent with real-world situations.

For comparison, the precision listed in [2] was measured in ideal laboratory conditions with all variables accounted for. In such unrealistic conditions our system will exhibit higher precision, due to the larger baseline.

D. Outdoor experiments of mutual localization of two UAVs

To verify the selectivity of the sensor, as well as the range and precision in an outdoor environment, which is the
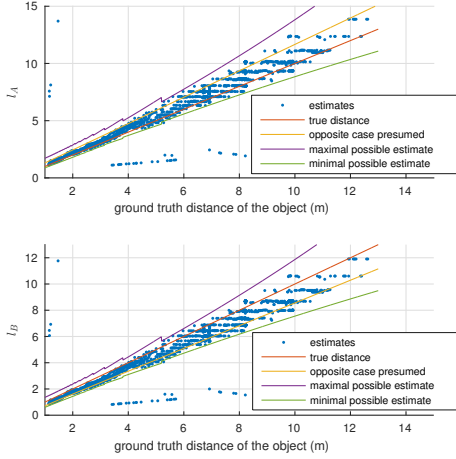
Fig. 13: The precision of estimation of the distance while presuming the orthogonal alignment (upper plot) and the 30 ° alignment (lower plot) of the observed points.
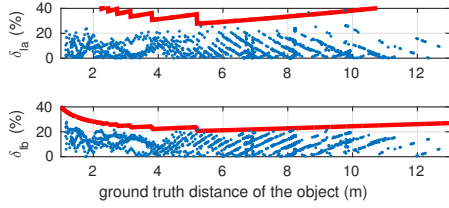

Fig. 14: Relative errors δ_{la} and δ_{lb} of the estimated distances in case of presuming the orthogonal alignment (upper plot) and the 30 °alignment (lower plot) respectively. The red line denotes predicted maximal relative error.

Fig. 15: UAV distance estimates compared with the ground truth. The two estimates according to equations (5) - case A and (6) - case B tend to be greater and smaller than the ground truth respectively. The spurious pattern around 60 s and 110 s is the result of the observed UAV spinning in place, leading to the two equations rapidly exchanging their validity.
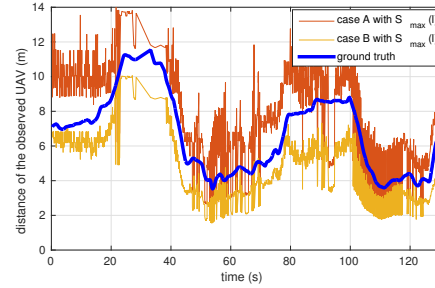
Fig. 16: The same estimates as in figure 15 with the angle α expanded by $k(S(l) - \frac{1}{2})$.

primary intended use-case, we have performed a flight with a pair of our outdoor UAV platforms. The platform itself is a DJI 550 frame, equipped with a Pixhawk controller an Intel NUC computer, described in detail in [26]. The sensory equipment comprises a range sensor for altitude measurements, RTK-GNSS antenna for ground truth measurement, the UV camera described in this paper and a color camera for a conventional video recording. The target UAV was equipped with six ultraviolet LED markers as described above. The exposure rate of the camera was set to 1000 µs.

The markers could still be detected at 15 m and after that there were spurious drop-offs. Compare the table III in [2], where the maximum range is 5.5 m for the highest resolution. Indeed, since the camera FoV listed in the paper is only 42°while our system uses a FoV of 180°, our system can be meaningfully compared with their system being used with less than the smallest listed resolution, where the maximum range was only 3.2 m.

An example of the image obtained by the ultraviolet-sensitive camera can be seen in the figure 2. We have attempted to measure the precision of the ultraviolet marker-based localization by using the RTK-GNSS data inertial measurement unit and built-in compass as ground truth. As is shown in figures 15 and 16, the distance estimations according to equations (5) and (6 resulted in upper and lower margins, enclosing the ground truth. While the position retrieved by RTK-GNSS was sufficiently precise, the orientation estimate was burdened by a severe drift (see figure 17-below), causing misalignment in the bearing vector returned by our UV sensor. This hints at an alternative application of our system as a precise orientation sensor if the position of two or more mutually unoccluded UAVs is known from RTK-GNSS data, since the observed relative bearing does not drift.

VI. CONCLUSION

In this paper we have proposed a novel system for outdoor and indoor mutual relative localization using ultraviolet LED makers. The main intended use-case of this method is in swarm control and stabilization of formations of light-weight helicopter UAVs in arbitrary environments. The proposed
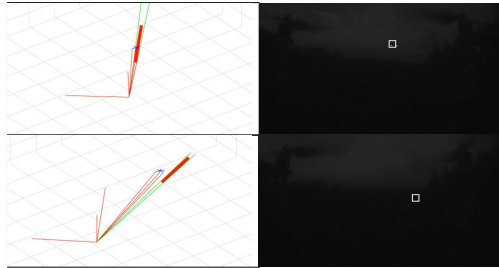
Fig. 17: Selected frames from the video output of the camera and visualization of the mutual state estimation procedure. The blue star and white square represent the ground truth of the observed UAV. The thick red line shows the range of possible positions computed by our system. This can be seen in a video found at `http://mrs.felk.cvut.cz/uvdd1`.

approach enables significantly higher detection range, robustness to light conditions and surrounding environment (background of images) in comparison with state-of-the-art methods, while having low computational intensity, small size and weight and providing sufficient precision. The error in the relative bearing vector towards a single ultraviolet marker is in operational distances below 0.02 rad in distances above 3 m, and thus allows direct applicability of the method using most of the current swarming and formation stabilization approaches. Two algorithms for estimating the mutual distance were developed to satisfy requirements of known multi-UAV stabilization techniques. The first relies on the size of the detected spot in the image, while the second is based on the diminishing apparent internal distance between a pair of retreating markers with known true mutual distance. The system was tested in outdoor environment and was shown to be robust with respect to outdoor lighting conditions as predicted. The theoretical predictions, as well as the experimental data presented here, show a lot of promise for deployment in swarm robotics.

REFERENCES

[1] M. Nagy, Z. Akos, D. Biro, and T. Vicsek, "Hierarchical group dynamics in pigeon flocks," *Nature*, vol. 464, no. 7290, p. 890, 2010.

[2] J. Faigl, T. Krajník, J. Chudoba, L. Preucil, and M. Saska, "Low-cost embedded system for relative localization in robotic swarms," in *IEEE ICRA*, 2013.

[3] T. Krajník, M. Nitsche, J. Faigl, P. Vaněk, M. Saska, L. Přeučil, T. Duckett, and M. Mejail, "A practical multirobot localization system," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 3-4, pp. 539–562, 2014.

[4] M. Saska, V. Vonasek, T. Krajnik, and L. Preucil, "Coordination and Navigation of Heterogeneous UAVs-UGVs Teams Localized by a Hawk-Eye Approach," *International Journal of Robotics Research*, vol. 33, no. 10, pp. 1393–1412, 2014.

[5] M. Saska, T. Báča, J. Thomas, J. Chudoba, L. Preucil, T. Krajník, 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.

[6] M. Saska, "Mav-swarms: Unmanned aerial vehicles stabilized along a given path using onboard relative localization," in *International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2015.

[7] 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 *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[8] A. Kushleyev, D. Mellinger, C. Powers, and V. Kumar, "Towards a swarm of agile micro quadrotors," *Autonomous Robots*, vol. 35, no. 4, 2013.

[9] P. Stegagno, M. Cognetti, G. Oriolo, H. H. Bülthoff, and A. Franchi, "Ground and aerial mutual localization using anonymous relative-bearing measurements," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1133–1151, Oct 2016.

[10] G. Vásárhelyi, C. Virágh, G. Somorjai, N. Tarcai, T. Szörényi, T. Nepusz, and T. Vicsek, "Outdoor flocking and formation flight with autonomous aerial robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2014*. IEEE, 2014.

[11] C. Virágh, G. Vásárhelyi, N. Tarcai, T. Szörényi, G. Somorjai, T. Nepusz, and T. Vicsek, "Flocking algorithm for autonomous flying robots," *Bioinspiration & biomimetics*, vol. 9, no. 2, 2014.

[12] G. Michieletto, A. Cenedese, and A. Franchi, "Bearing rigidity theory in SE(3)," in *55th IEEE Conf. on Decision and Control*, Las Vegas, NV, Dec. 2016, pp. 5950–5955.

[13] A. Zunino, M. Crocco, S. Martelli, A. Trucco, A. D. Bue, and V. Murino, "Seeing the sound: A new multimodal imaging device for computer vision," in *IEEE ICCVW*, 2015.

[14] X. Wang, Y. Sekercioglu, and T. Drummond, "Vision-based cooperative pose estimation for localization in multi-robot systems equipped with rgb-d cameras," *Robotics*, vol. 4, no. 4, p. 122, Dec 2014.

[15] I. Senthooran, J. C. Barca, and H. Chung, "A 3d line alignment method for loop closure and mutual localisation in limited resourced mavs," in *ICARCV*, 2016.

[16] K. Boudjit and C. Larbes, "Detection and target tracking with a quadrotor using fuzzy logic," in *8th International Conference on Modelling, Identification and Control (ICMIC)*, 2016.

[17] V. Dhiman, J. Ryde, and J. J. Corso, "Mutual localization: Two camera relative 6-dof pose estimation from reciprocal fiducial observation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013.

[18] R. Tron, J. Thomas, G. Loianno, J. Polin, V. Kumar, and K. Daniilidis, "Vision-based formation control of aerial vehicles," in *Robotics: Science and Systems*, 2014.

[19] I. Rekleitis, P. Babin, A. DePriest, S. Das, O. Falardeau, O. Dugas, and P. Giguere, "Experiments in quadrotor formation flying using on-board relative localization (technical report)." [Online]. Available: http://www2.ift.ulaval.ca/~pgiguere/papers/ARdroneCL_Workshop.2015.pdf

[20] 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 *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013.

[21] D. Brandtner and M. Saska, "Coherent swarming of unmanned micro aerial vehicles with minimum computational and communication requirements," in *European Conference on Mobile Robots (ECMR)*, 2017.

[22] A. Franchi, C. Masone, V. Grabe, M. Ryll, H. H. Bülthoff, and P. R. Giordano, "Modeling and control of uav bearing formations with bilateral high-level steering," *The International Journal of Robotics Research*, vol. 31, no. 12, 2012.

[23] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A flexible technique for accurate omnidirectional camera calibration and structure from motion," in *Fourth IEEE International Conference on Computer Vision Systems (ICVS'06)*, 2006.

[24] D. Scaramuzza. (2018) Ocamcalib: Omnidirectional camera calibration toolbox for matlab. Accessed on 02-18-2018. [Online]. Available: https://sites.google.com/site/scarabotix/ocamcalib-toolbox

[25] P. Stegagno, M. Cognetti, A. Franchi, and G. Oriolo, "Mutual localization using anonymous bearing measurements," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011.

[26] G. Loianno, V. Spurny, T. Baca, J. Thomas, D. Thakur, D. Hert, R. Penicka, T. Krajnik, A. Zhou, A. Cho, M. Saska, and V. Kumar, "Localization, grasping, and transportation of magnetic objects by a team of mavs in challenging desert like environments," *IEEE Robotics and Automation Letters*, 2018, (accepted to RA-L and ICRA). [Online]. Available: http://ieeexplore.ieee.org/document/8276269/

[27] M. Saska, J. Vakula, and L. Preucil, "Swarms of Micro Aerial Vehicles Stabilized Under a Visual Relative Localization," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.

Mutual Localization of UAVs based on Blinking Ultraviolet Markers and 3D Time-Position Hough Transform

Viktor Walter[1], Nicolas Staub[1,2], Martin Saska[1] and Antonio Franchi[2]

Abstract— A novel vision-based approach for indoor/outdoor mutual localization on *Unmanned Aerial Vehicles* (UAVs) with low computational requirements and without external infrastructure is proposed in this paper. The proposed solution exploits the low natural emissions in the near-*Ultra-Violet* (UV) spectrum to avoid major drawbacks of the visible spectrum.Such approach provides much better reliability while being less computationally intensive. Working in near-UV requires active markers, which can be leveraged by enriching the information content through blinking patterns encoded marker-ID. In order to track the markers motion and identify their blinking frequency, we propose an innovative use of three dimensional Hough Transform, applied to stored position-time points. The proposed method was intensively tested onboard multi-UAV systems in real-world scenarios that are very challenging for visible-spectrum methods.The results of our methods in terms of robustness, reliability and precision, as well as the low requirement on the system deployment, predestine this method to be an enabling technology for using swarms of UAVs.

I. INTRODUCTION

The use of swarms of *Unmanned Aerial Vehicle*s (UAVs) has extends significantly the capabilities of single UAV, allowing for tasks otherwise impossible for single robot due to payload, actuation or sensory limitations. Typically, small UAVs are used for their cost-effectiveness and commercial availability and they can safely compose compact multi-UAV systems with small relative mutual distances. This raises the importance of mutual relative localization, in order to maintain safety distances, enforce the desired flocking behavior or the decentralized bio-inspired swarm stabilization [1], [2].

A typical challenge of mutual localization for aerial swarms is to present a low-cost infrastructure-independent solution, suitable for both indoor and outdoor settings and reasonable mutual distances. Literature is rich in approaches relaxing these requirements, like indoor work conducted with *Motion Capture systems* (MoCap), *e.g.* [3], [4], or *Infrared* blinking markers, coupled with an *event-based* ground camera [5] and outdoor setups relying on *Global Navigation Satellite System* (GNSS) [6], [7]. These solutions provide precise mutual localization information (≤ 1 cm, considering *RTK-GNSS*) with the major drawback of requiring pre-installed infrastructures, limiting the usage to known, uncluttered, and easily-accessible environments. Additionally, they are costly and tend to rely on intensive radio-communication
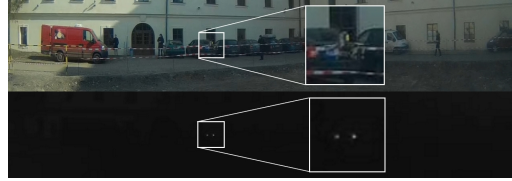
Fig. 1: Far away UAV against urban background in the shade. Barely noticeable in visible spectrum but obvious in UV spectrum.

between the swarm members, which is subject to limited range, interferences and does not scale up for large swarms.

Typical solutions to these issues are visible-spectrum vision-based approaches [8]. In an indoor-only setup, color-based markers can be used, see [9], [10], that are easy to segment under controlled lighting conditions, but not in the extremely unpredictable lighting conditions and the multi-colored outdoor environment. For outdoors, black and white markers are preferred, leading to solutions which combine passive markers and object detection, see [11], [12], used for swarms [2], [13] and heterogeneous groups of robots [14], [15], [16]. The drawbacks of these approaches are the need for large markers, computational complexity and sensitivity to lighting conditions.

Our solution extends our previous research on a novel, vision-based mutual localization in the Ultra-Violet (UV) spectrum [17] . Motivated by the low amount of near-UV radiation in sunlight and most artificial sources, compared to the visible spectrum. The technology uses active UV markers and standard cameras with UV band-pass filters, allowing for fast detection of markers in complex environments.

In order to retrieve the orientation or identity (ID) information, we encoded individual marker ID in blinking patterns. These are retrieved using an unprecedented application of 3D Time-Position Hough Transform. Indeed, the algorithm presented is the first exploiting the Hough Transform for tracking of objects in time. Since, in this case, the precision of the shape fitting is less relevant than the computational speed, this is en exemplary use-case for such algorithm.

The rest of the paper is structured as follows. Sec. II introduces the theoretical background necessary for the proposed algorithm presented in Sec. III. Finally, Sec. IV summarizes the results of our experimental proof-of-concept.

II. THEORETICAL BACKGROUND

A. UV spectrum: properties and motivations

For the sake of brevity, only key properties for our approach are presented, more details are presented in [17].

[1] CTU in Prague, FEE, Department of Cybernetics, Czech Republic
`{viktor.walter|martin.saska}@fel.cvut.cz`
[2] LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France
`{nicolas.staub|antonio.franchi}@laas.fr`

The solar spectrum approximates the *black-body radiation model* and has its peak intensity in the visible spectrum [18], while UV radiation is significantly less intense even close to the visible spectrum. This can be leveraged using affordable near-UV LEDs and suitable band-pass filters applied on a monochromatic camera. Tests have shown that the UV radiation refracted through the atmosphere and reflected from matte surfaces can be neglected, see [17]. Artificial radiation sources with strong UV emissions are rare, making this wavelength range very attractive for our application.

The image spots caused by UV LEDs sources are saturated at any time of the day, while the only other bright features are the sun or its specular reflections. Such images can be binarized by *static thresholding*, as opposed to the more computationally intensive *adaptive thresholding* needed for the visible spectrum. This allows for computationally simple detection of markers in the UV-range, robust to outdoor lighting conditions[1].

Following extensive testing in the wavelengths range of commercial UV LEDs. Best results are achieved with near-UV wavelength of 395 nm, compatible with widely available fisheye lenses with a large *Field of View* (FoV).

B. 3D time-position parametrization

With the proposed near-UV vision system, active markers appear in the camera image as white spots against a nearly-black background, see Fig. 1. While easy to locate, they are individually anonymous. In order to enrich their information content we devised blinking patterns encoding ID information (single marker ID or ID common to markers on one side of a UAV). While a non-blinking marker can be tracked among two consecutive camera frames using the nearest distance between two frames, this is not possible for blinking markers which are only visible in their *on-frames*, and periodically disappear in their *off-frames*. The only impact of this addition is a decrease of the *admissible flight dynamics*. On the other hand the introduction of blinking patterns does not only provide additional information, but also increases robustness by allowing 1) to easily filter out the sun and its reflections (non-blinking bright spots) and 2) to detect aliasing or occlusion, if the detected blinking frequency is not among the set of given patterns.

Identification and tracking of these blinking markers calls for a fast algorithm able to accommodate their periodical disappearance, which will use the following time-position parametrization. First, consider that markers observed by a camera are defined by their *x-y* coordinates in the image plane. These can be stored in an accumulator along time, with *t-indexes* such that the latest camera image corresponds to $t = 0$. We refer to triplets (x, y, t) as *t-points* which correspond to observations. Then for a set of markers with limited physical dynamics, their t-points will lie along smooth curves w.r.t. time, intermittent for blinking markers, see Fig. 2-a.

We chose to approximate those curves, around $t = 0$, by lines, see Fig. 2-b. We refer to these lines as *t-lines*. The time window used for this approximation impacts both the admissible flight dynamics and the range of usable blinking
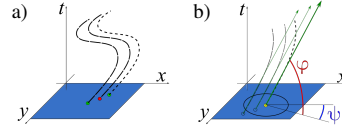

Fig. 2: Basic assumptions for the proposed system. (a) moving points (green) in present camera images (blue plane) follow smooth trajectories w.r.t. time. Due to blinking of the markers these curves are intermittent, making some of the points temporarily invisible (red). (b) considering short enough time-span these curves can be approximated by lines and such lines can be parametrized by their origin-point (yellow) and their pitch φ and yaw ψ.

frequencies. The t-lines can be parametrized by their *origin-point x-y* coordinates and two angles that we call pitch φ and yaw ψ, see Fig. 2-b. The origin-points are the points located in the image plane where the t-lines intersect. In on-frames they coincide with a t-point of $t = 0$, otherwise they are retrieved via Hough Transform, as detailed in Sec. III, and correspond to the theoretical marker position along the curve. The pitch and yaw map to the image speed of the tracked marker, and to the direction of its motion respectively.

Clearly the blinking frequency of a tracked marker can be retrieved via the t-points spacing along a t-line, the construction of which relies on a Hough Transform, see Sec. III. An additional benefit of considering t-lines is that markers fixed to a translating rigid body will have parallel t-lines, thus allowing for association of markers and objects.

C. Hough Transform

Hough Transform is a well-known method used to retrieve regular geometric forms described by a set of parameters. An example application is fitting a over a set of collinear points. This is done by projecting the points into *image matrix* in the form of curves representing the range of possible lines passing through this point in terms of their parameters. The pixels of the temporary image matrix (Hough space) are incremented along each of these curves. Such projections of a number of collinear points will intersect in the Hough space, creating a local maximum in value representing the most likely parameters of a line common to all the original points. For general 3D line fitting scenarios with reasonable precision the Hough Transform will require a dense discretization of the parameter space consisting of at least four parameters [19]. This means searching for local maxima in a large 4D space which is not feasible for UAV embedded solutions. Instead we use a more purpose-fitted implementation of Hough Transform, relaxing t-lines reconstruction to guarantee a good enough approximation in order to reliably separate adjacent markers. As we aim at a good enough approximation, the discretization steps of the t-line parameters, $\Delta\varphi$ and $\Delta\psi$, can be chosen large enough to enforce robustness against small errors in the origin-point coordinates arising from the camera image pre-processing. Moreover the set of possible t-lines is constrained by the physics of the system, allowing to reduce the size of the Hough space.

[1]`mrs.felk.cvut.cz/uvdd1`
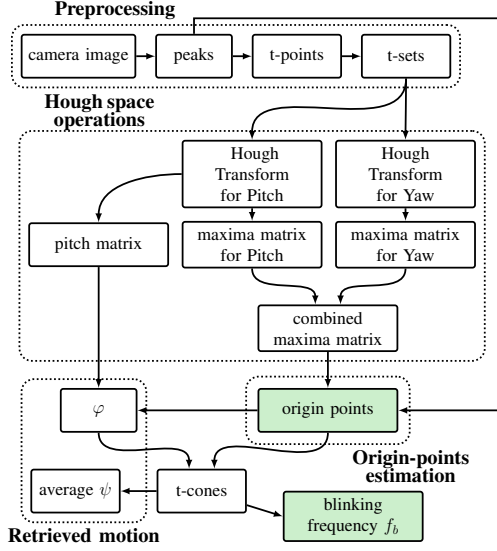
Preprocessing

Fig. 3: Flow chart of our tracking algorithm based on Hough Transform. Both origin-point and blinking frequencies are retrieved, by-products are the full t-line parameters.

Fig. 4: Erroneous peaks (red) in Hough Transform for pitch (left) and yaw (center), are suppressed by element-wise multiplication.

in the Hough spaces will intersect at voxels corresponding to parameters of the t-line on which they lie.

To find the t-line parameters one needs to find local maxima in a 3D voxel space, which is computationally complex. We use the fact that since the curves followed by the t-points are non-retreating w.r.t. time and since the markers are attached around non-transparent objects, it is physically impossible for multiple t-lines to share the same origin-point. With this assumption, we can simplify the search for local maxima into 2D, the Hough spaces are flattened into so-called *maxima matrices*. This operation is done by assigning to each $[x, y]$ element of the maxima matrix the highest value among voxels of the Hough space with the same *x-y* coordinates. This results in easier detection for origin-coordinates at the expense of an information loss about the associated angle parameters. To keep this information easily accessible, a second matrix, so-called *angle matrix*, is constructed during the flattening process, which stores the angle value (respectively ψ or φ) corresponding of the maxima in the Hough space for each *x-y* coordinates.

3) Origin-point retrieving: From there origin-points can be retrieved as peaks in the maxima matrices. However some aliasing phenomena, as well as ambiguity-based artifacts may be found in the maxima matrix. For pitch, erroneous peaks in between two slow moving markers can appear. For yaw, two different kinds of erroneous peaks can appear; some corresponding to opposite yaw, as well as peaks perpendicular to the connecting lines of neighboring markers due to discretization. To increase robustness against these erroneous peaks, the two maxima matrices are multiplied element-wise, which leads to the suppression of the erroneous peaks as they are not likely to be present in both spaces.

The origin-point coordinates correspond to peaks in the combined maxima matrix, and are retrieved in a two-step method; 1) t-point of index t=0 are collected as their coordinate are more reliable than the estimated one, then 2) peaks in the combined maxima matrix are collected. After each located peak, its surroundings are nullified in the combined maxima matrix, allowing for finding further peaks. During the peak search we also consider the number of expected origin-points, L, once L peaks are found the search can stop. The policy to define L can be based on considering: 1) the knowledge of visible origin-points based on the number of UAVs and the average number of visible markers, 2) the maximum number of markers seen simultaneously within the last F frames or 3) any other heuristic.

At this stage of the algorithm, the t-line origin-points are retrieved, *i.e.* the image positions of markers both in on- and off-frames. From the respective angle matrices, it is possible to retrieve the two other t-line parameters estimates.

4) Blinking frequency retrieval: As we decided to encode additional information into blinking patterns further processing is needed to retrieve them. A possible way to do so

III. ALGORITHM FOR ORIGIN-POINT POSITION AND BLINKING FREQUENCY TRACKING

This section details the proposed algorithm to retrieve the origin-points and their blinking frequencies from the camera grey-scale image. The overall flow is summarized in Fig. 3.

A. Base algorithm

1) Image pre-processing: The pre-processing of the grey-scale image starts by the detection of bright spots, which are used to construct t-points, see Sec. II-B. The t-points are stored in a set, $\mathcal{U} \subset \mathbb{N}^3$. These t-points can be interpreted as points in a bounded 3D space of height F corresponding to the highest t-index, *i.e.* the time window of $\mathcal{U}$. The set of t-points is updated with each new camera image. We are interested in a way to retrieve for each t-line both the origin-point and the blinking frequency along the t-line.

2) Hough Space Operations: A direct approach would consist in applying 4D Hough Transform on $\mathcal{U}$ directly, which proves to be computationally expensive and cumbersome. Therefore, our algorithm is based on two simpler 3D Hough voxel spaces, considering origin-point coordinates combined with pitch and yaw separately. To reduce the search space for the construction of the Hough Transform, the pitch and yaw are discretized such that

$$\psi_j = j \frac{2\pi}{\Delta\psi} \mid j \in [\frac{\underline{\psi}}{\Delta\psi}; \frac{\overline{\psi}}{\Delta\psi}] \subset \mathbb{N} \quad \text{and} \quad \varphi_i = i \frac{\pi}{2\Delta\varphi} \mid i \in [\frac{\underline{\varphi}}{\Delta\varphi}; \frac{\overline{\varphi}}{\Delta\varphi}] \subset \mathbb{N},$$

where the range limits and discretization steps are parameters of our algorithm, listed in Tab. I. Our specialized Hough Transform translates t-points into their images in the form of voxelated surfaces in the two aforementioned voxel spaces. If multiple t-points belong to the same marker, their images
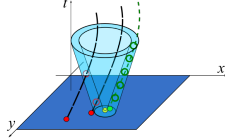
Fig. 5: Cone shell defined by the estimated t-line pitch. The t-lines with distant origin-point (red filled) will intersect the expanded cone shell in few points (hollow red), while t-lines with origin-point (green filled) nearby to the center of the t-cone (yellow) will intersect in most of their points (hollow green). This suppresses their influence on the estimated frequency and yaw.

is to cluster all t-points in the vicinity of the identified t-lines to find their average blinking frequency. To reduce computational requirements induced by exploring two Hough spaces of small granularity, we choose $\Delta\varphi \ll \Delta\psi$ and forgo an angle matrix for ψ. This only gives a reliable estimate of the pitch parameter, while the ψ maxima matrix is still rich enough to reject aliasing. Instead, we now consider t-cones, which are generated by t-lines rotated around their t-axis passing through the origin point, see Fig. 5, and their vicinity to retrieve both blinking frequency and t-line yaw, by averaging the values of the corresponding t-points. The vicinity r_v is a tunable parameter defining the maximum distance from the t-cone where we look for t-points. Note that this method is likely to collect more t-points which are not a part of the desired t-line in the averaging process, see Fig. 5. Nevertheless they are, in practice, outnumbered by the t-points corresponding to the desired t-line. Lastly, the origin-point with blinking frequency under a certain (low) threshold can be disregarded as being the sun or its specular reflections, which are the most significant contaminant of the camera image in the UV spectrum.

This concludes the algorithm as both origin-points and their associated blinking frequency are retrieved, along with the other t-lines parameters.

B. Improvements

In order to increase the robustness to high flight dynamics and computational efficiency, we have designed two refinements of our algorithm.

1) Weighted Hough space: First, to increase the admissible dynamics of the tracked markers we propose to introduce weight in the construction of the Hough space. Instead of giving equivalent weight for all t-indices, we introduce the following weighting function,

$$w(t) = \lambda(F - t) + F \in \mathbb{N},$$

where λ is a parameter regulating the weight ratio between the newest t-points, and the oldest ones in $\mathcal{U}$. In this way the more recent t-points affect the t-lines parameters more, making our algorithm more resilient to abrupt changes of direction, implied by highly dynamic flight. This refinement leads directly to better origin-point estimate.

2) Pre-computed masks: The second refinement reduces the computational complexity of the Hough space construction, by applying pre-computed masks to generate it. The constructed masks resemble hollow cones for the pitch
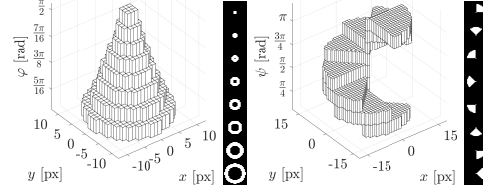

Fig. 6: A mask used in Hough space for pitch, generated for t-points with $t = 10$ (left) and a mask for yaw, generated for t-points with $t = 15$ (right). Side bitmaps show the corresponding mask slices.

Hough space, see Fig. 6-left, and spiral staircases for the yaw Hough space, see Fig. 6-right. These shapes can be explained intuitively. We observed that the possible origin-points of all t-lines passing through a t-point, can be easily expressed w.r.t. the t-line parameters, φ and ψ, for a given t-index t. The potential origin-point for this t-point directly underneath has the parameter $\varphi = \frac{\pi}{2}$ and as the distance of the potential origin-points increases, the corresponding pitch decreases, which leads to a cone shape in the Hough space. Such cones are of the same shape for a given t, while the x and y parameters of the t-point merely shift it to the respective x-y position. Similar reasoning explains the mask shape for the yaw Hough space. In order to prevent discontinuities in the masks, which arise from the angle discretization, we introduce overlap parameters, $\delta\varphi$ and $\delta\psi$.

To construct the Hough space, at each t-point in $\mathcal{U}$, the introduced masks are used as follow: 1) retrieve the mask associated with the t-index 2) retrieve the x-y coordinates of the t-point and then 3) apply the mask at the coordinates by increasing the corresponding voxel values in the Hough space. This considerably speeds up the construction of the Hough space as instead of calculating all t-lines passing through each t-point we apply a static, pre-computed, mask.

IV. EXPERIMENTAL VALIDATION

Experimental parameters are grouped in Tab. I. The chosen parameters influence the maximal admissible flight dynamics, in our case the maximum linear image speed for a marker is 144 px/s, which translates to a maximum speed perpendicular to the camera axis of respectively 0.6 and 3 m s^{-1}, for marker-camera distance of 1 and 5 m.

The innovative part of the system, the UV active markers and camera, are introduced and discussed in [17]. They can be easily fitted to any UAV platform, which we demonstrate by using a standard hexarotor for outdoor experiments and a standard quadrotor for the indoor ones. Indoor experiment used motion capture system (MoCap) as ground truth.

In the testing phase, our algorithm runs real-time off-board on an Intel NUC 7 (4 cores, 2.6 GHz), a classical embedded computer for UAVs. The prototypical MATLAB implementation loads a single core up to 48.8% on average, validating that our approach can be easily embedded. Inded there are enough resources left to run typical mission control and planning algorithms. The final C code implementation is expected to run significantly faster. The experiments were recorded on video (`mrs.felk.cvut.cz/uvdd2`).

Symbol	Meaning	Impact on	Experimental Value [unit]
F	time window size of $\mathcal{U}$	admissible dynamics	$0.3f$
$\Delta\varphi$	φ discretization step in Hough Transform	robustness to dynamic motion; selectivity; computational complexity	$\pi/64$ rad
$\Delta\psi$	ψ discretization step in Hough Transform	computational load	$\pi/4$ rad
$\delta\varphi$	mask overlap parameter	robustness; overall selectivity	$\pi/68$ rad
$\overline{\varphi}$	upper limit of the pitch value	minimal expected movement	$\pi/2$ rad
$\underline{\varphi}$	lower limit of the pitch value	admissible dynamics; marker selectivity (via $\Delta\varphi$)	$\pi/4$ rad
L	number of expected peaks	number of markers; computational complexity	3 (o) 10 (i)
λ	weighting factor	admissible dynamics	1
r_v	inspected vicinity of t-cones	robustness; marker selectivity	3 px
f	camera frame-rate	maximal blinking frequency $\overline{f_b}$	72 Hz
t_x	camera exposure rate	selectivity w.r.t. ambient radiation; effective distance range	1000 µs
$[\underline{f_b};\overline{f_b}]$	blinking frequency range	number of ID encodable	3.34 – 40 Hz

TABLE I: Main parameters of the proposed solution, when needed (o) and (i) denotes outdoor and indoor parameters respectively.
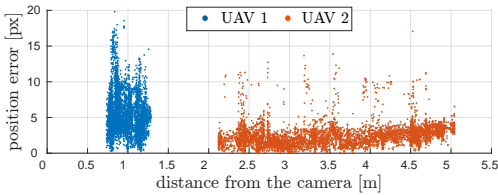

Fig. 7: Indoor precision testing with MoCap as ground truth, the position error in the camera horizontal direction against the distance to the camera is assessed. Marker close have larger (but reasonable) error, due to larger spot in image, making the exact pixel position of the marker ambiguous.

A. Indoor validation against ground truth

To evaluate the accuracy, two quadrotor UAVs are flown in front of a fixed camera, markers are located on the UAV arms, as described in [17]. To identify each UAV, blinking patterns are assigned such that each UAV has two dedicated frequencies, one for front and one for back markers.

In the experiment the *UAV 1* was hovering within 1 m of the camera while performing yaw rotation motion, with blinking frequencies of 6 and 10 Hz. The *UAV 2* was following a zig-zag like trajectory from 5 to 2 m toward the camera, with blinking frequencies of 15 and 30 Hz. The UAVs motion are constrained by the limited size of the flying arena, forbidding tests on distance longer than 5 m.

MoCap information is translated to camera image and MoCap-based image positions are paired with the closest estimated positions of origin points.The position error against the camera-marker distance is used to asses performances, see Fig. 7. Markers close to the camera suffer an error of at most 20 px, while for further away markers the error is mostly below 5 px. This correlates with the size of the bright spots in the image that make t-point detection less precise.

The results were additionally evaluated w.r.t. the blinking frequency of the markers, see Fig. 8. It appears that the blinking frequencies have not detectable influence on the position error, as the trend for each individual frequency follows the aggregated values for their respective UAV.

B. Outdoor validations and characterization

Additional outdoor experiments were conducted to assess our approach performances in operational conditions. Experiments were conducted around noon by clear weather and consisted of flights of two hexarotor UAVs, one equipped
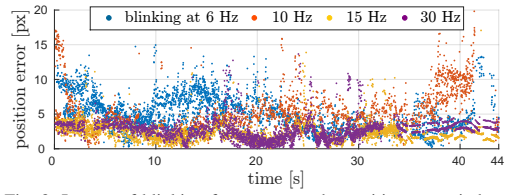
Fig. 8: Impact of blinking frequency on the position error, indoor.
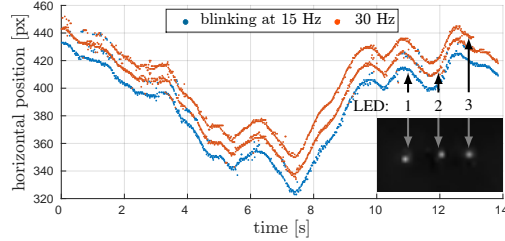
Fig. 9: Example of outdoor tracking for UAV horizontal motion (the most agile). Near the instant $t = 13$ s the UAV was rotated, so that only two markers remained visible.

with a camera and one equipped with markers following [17]. The markers were set to blink with two distinct frequencies, such that two triplets on adjacent hexarotor arms shared frequency. Blinking frequencies where set at 15 and 30 Hz, for back and front respectively. The tracking results presented in Fig. 9 show the good performance of the proposed approach under outdoor light conditions.

In particular, our algorithm was able to keep track of the IDs of the markers encoded in their blinking frequency, while still providing accurate image position estimation even in between the on-frame t-points.

C. Blinking frequency estimation

Both indoor and outdoor data are used to assess the performances of the frequency estimation, as the generated frequency are known, they are compared in Fig. 10. By filtering out the obvious outliers, we compute the average error for all points close to a given frequency. Performances of the blinking frequency estimation, both indoor and outdoor, are good with mean absolute error (MEA) below 3.9%, 2.2%, 3.8% and 3.1% for respectively 6, 10, 15 and 30 Hz blinking frequency. This also demonstrates that the blinking frequency

Method	Resolution	FoV	Range	ID count	Environment	Properties
Color circles [20]	752×480	$125°$	N/A	3+	indoor, well illuminated	large marker size, lighting sensitive
WhyCon [11]	752×480	$42°$	5.5 m	1	indoor/outdoor, illuminated	large marker size (≤ 18 cm)
ALM-DVS [5]	128×128	$65°$	N/A	3+	indoor	requires event-based camera
CNN-YOLO [21]	1280×720	$132°$	15 m	N/A	indoor/outdoor, illuminated	high computational load, marker independant
Proposed approach	752×480	$180°$	**15** m	**6+**	**indoor/outdoor**	**small markers, low computational intensity**

TABLE II: Performance comparison with various representative methods.

Fig. 10: Estimated Frequencies evolution, in both indoor (top) and outdoor (bottom) experiments. In both cases the frequency estimation performs with a MEA below 3.9%.

estimation performs similarly for all frequencies well inside the admissible blinking frequency range.

D. Comparison with other methods

Our proposed methods is significantly more versatile than state-of-the-art methods, allowing usage both indoor and outdoor without illumination requ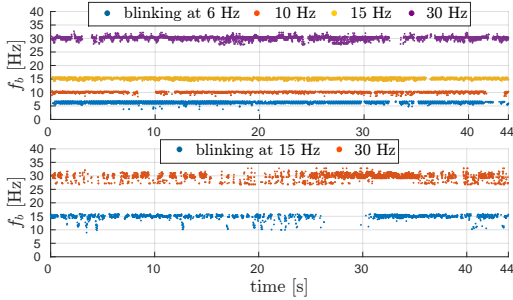irements, see Tab.II. Moreover our appraoch has a low impact in terms of computational power and works with small markers. For each method the precision and range can be tuned by selecting different resolution and FoV, therefore there are no standard metrics to evaluate them. Despite the combination of small resolution and large FoV in our experiments, the performances are comparable or exceeding those of other methods.

V. Conclusion

In this paper, we proposed a novel system for outdoor and indoor mutual relative localization using active UV LED markers. It enables significantly better performances in comparison with state-of-the-art methods of UAV mutual localization. Additionally, we have shown how active markers can be leveraged to encode additional information via blinking patterns. Our approach relies on 3D time-position Hough Transform and has been tested in active UAV deployment both indoor and outdoor. Results from outdoor experiments show excellent detection reliability w.r.t. backgrounds such as the sky, trees or even buildings, while still being able to decode the blinking signal. The theoretical predictions, as well as the experimental data presented here, show a lot of promise for deployment in swarm robotics and multi-robot systems in general.

[2] M. Saska, "Mav-swarms: Unmanned aerial vehicles stabilized along a given path using onboard relative localization," in *International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2015.

References

[1] D. Brandtner and M. Saska, "Coherent swarming of unmanned micro aerial vehicles with minimum computational and communication requirements," in *European Conference on Mobile Robots (ECMR)*, 2017.

[3] A. Kushleyev, D. Mellinger, C. Powers, and V. Kumar, "Towards a swarm of agile micro quadrotors," *Autonomous Robots*, vol. 35, no. 4, 2013.

[4] P. Stegagno, M. Cognetti, G. Oriolo, H. H. Bülthoff, and A. Franchi, "Ground and aerial mutual localization using anonymous relative-bearing measurements," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1133–1151, Oct 2016.

[5] 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 *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013.

[6] G. Vásárhelyi, C. Virágh, G. Somorjai, N. Tarcai, T. Szörényi, T. Nepusz, and T. Vicsek, "Outdoor flocking and formation flight with autonomous aerial robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2014*. IEEE, 2014.

[7] C. Virágh, G. Vásárhelyi, N. Tarcai, T. Szörényi, G. Somorjai, T. Nepusz, and T. Vicsek, "Flocking algorithm for autonomous flying robots," *Bioinspiration & biomimetics*, vol. 9, no. 2, 2014.

[8] M. Nagy, Z. Akos, D. Biro, and T. Vicsek, "Hierarchical group dynamics in pigeon flocks," *Nature*, vol. 464, no. 7290, p. 890, 2010.

[9] R. Tron, J. Thomas, G. Loianno, J. Polin, V. Kumar, and K. Daniilidis, "Vision-based formation control of aerial vehicles," in *Robotics: Science and Systems*, 2014.

[10] I. Rekleitis, P. Babin, A. DePriest, S. Das, O. Falardeau, O. Dugas, and P. Giguere, "Experiments in quadrotor formation flying using on-board relative localization (technical report)."

[11] J. Faigl, T. Krajník, J. Chudoba, L. Preucil, and M. Saska, "Low-cost embedded system for relative localization in robotic swarms," in *IEEE ICRA*, 2013.

[12] T. Krajník, M. Nitsche, J. Faigl, P. Vaněk, M. Saska, L. Přeučil, T. Duckett, and M. Mejail, "A practical multirobot localization system," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 3-4, pp. 539–562, 2014.

[13] M. Saska, T. Báča, J. Thomas, J. Chudoba, L. Preucil, T. Krajník, 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.

[14] M. Saska, V. Vonasek, T. Krajnik, and L. Preucil, "Coordination and Navigation of Heterogeneous UAVs-UGVs Teams Localized by a Hawk-Eye Approach," *International Journal of Robotics Research*, vol. 33, no. 10, pp. 1393–1412, 2014.

[15] K. Boudjit and C. Larbes, "Detection and target tracking with a quadrotor using fuzzy logic," in *8th International Conference on Modelling, Identification and Control (ICMIC)*, 2016.

[16] V. Dhiman, J. Ryde, and J. J. Corso, "Mutual localization: Two camera relative 6-dof pose estimation from reciprocal fiducial observation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013.

[17] V. Walter, M. Saska, and A. Franchi, "Fast mutual relative localization of uavs using ultraviolet led markers," in *2018 International Conference of Unmanned Aircraft System (ICUAS)*, 2018, accepted to ICUAS 2018.

[18] M. Iqbal, *An Introduction To Solar Radiation, Chapter 3: The Solar Constant and Its Spectral Distribution*, 1983.

[19] C. Dalitz, T. Schramke, and M. Jeltsch, "Iterative hough transform for line detection in 3d point clouds," *Image Processing On Line*, vol. 7, pp. 184–196, 2017.

[20] R. Tron, J. Thomas, G. Loianno, K. Daniilidis, and V. Kumar, "A distributed optimization framework for localization and formation control: applications to vision-based measurements," *IEEE Control Systems*, vol. 36, no. 4, pp. 22–44, 2016.

[21] M. Vrba, "Relative localization of helicopters from an onboardcamera image using neural networks," Master's thesis, Czech Technical University, Prague, 2018.

Optical communication-based identification for multi-UAV systems: theory and practice

Daniel Bonilla Licea[1*], Viktor Walter[1], Mounir Ghogho[2] and Martin Saska[1]

[1]Czech Technical University in Prague, Czechia.
[2]International Unversity of Rabat, Morocco.

*Corresponding author(s). E-mail(s): bonildan@fel.cvut.cz;
Contributing authors: viktor.walter@fel.cvut.cz; mounir.ghogho@uir.ac.ma;
martin.saska@fel.cvut.cz;

Abstract

Mutual relative localization and identification are important features for multi-Unmanned Aerial Vehicle (UAV) systems. Camera-based communications technology, also known as Optical Camera Communications (OCC) in the literature, is a novel technology that brings a valuable solution to this task. In such a system, the UAVs are equipped with LEDs acting as beacons, and with cameras to locate the LEDs of the other UAVs. Specific blinking sequences are assigned to the LEDs of each of the UAVs to uniquely identify them. This camera-based system is immune to Radio Frequency (RF) electromagnetic interference and operates in Global Navigation Satellite (GNSS)-denied environments. In addition, the implementation of this system is inexpensive. In this article, we study in detail the capacity of this system and its limitations. Furthermore, we show how to construct blinking sequences for UAV LEDs to improve system performance. Finally, experimental results are presented to corroborate the analytical derivations.

Keywords: mutual identification, multi-UAV system, OCC

1 Introduction

Mutual relative localization and identification are important features in multi-UAV. While relative localization is important for close cooperative flying and mutual collision avoidance, the identification of neighboring team members is crucial for high-level planning. This feature can be implemented using RF electromagnetic signals, vision-based techniques, or through a combination of both. For instance, in [1, 2], RTK-GNSS is used for UAV localization. In [3], Ultra Wide-band (UWB) ranging is used to determine the distance between the UAVs. In [4], they use a motion capture system

that sends its position estimate to the UAV via an RF link. Such localization techniques based on RF are vulnerable to electromagnetic interference and may fail in providing mutual identification in multi-robot systems. On the other hand, vision-based techniques are immune to electromagnetic interference and can provide relative localization and identification in multi-robot systems.

Vision-based localization and identification systems can be divided into passive and active, distinguished based on whether the optical markers emit light. In [5], the authors present a passive system where specific marker patterns are assigned to each robot. The same system was used

1

tion for multi-UAV systems: theory and practice

in [6] for outdoors localization and identification of UAVs. A disadvantage of passive systems is their sensitivity to ambient light, making them ineffective in poorly illuminated environments. This is solved by using active systems where the markers of UAVs are generally implemented with LEDs.

In [7], the authors equipped a UAV with infrared LEDs and used a CMOS camera to perform indoors localization. Different blinking frequencies, in the range of 1-2kHz, were assigned to each LED to differentiate them. The blinking frequencies were set in such a way that no two signals shared common harmonics to avoid ambiguities in their discrimination. In [8], our research group presented the UltraViolet Direction And Ranging (UVDAR) system for UAVs, see Fig. 1. In this system, UAVs are equipped with Ultraviolet (UV) LEDs as markers and cameras coupled with optical UV bandpass filters. The optical signals emitted by the LEDs were square signals of different frequencies. This is a simple way to discriminate the different blinking patterns, but it is inefficient as will be shown in this paper.

In the field of communications, the Optical Camera Communications (OCC) system, in which the transmitter is a LED and the receiver is a camera [9, 10], has been used for car-to-car communications and car-to-infrastructure communications [11, 12], and recently it has started to being applied to communications with UAVs [13, 14]. Despite the fact that the purpose of the OCC system is to exchange information, it can be used to improve the active vision-based localization and identification systems for UAVs.

This paper focuses on the mutual identification capacity of a vision-based active system, as the relative localization issue is beyond its scope. We investigate the mutual identification using the UVDAR system, as mentioned above.

The main contributions of this article are as follows:

- Blinking sequence generation method: a theoretical framework was developed to design sets of blinking sequences for the LEDs of UAV groups. These sequence sets are optimized to discriminate between as many sequences as possible in the shortest time. This enables large groups of UAVs to perform mutual identification in the shortest time.

- Theoretical analysis of UVDAR: we performed an analytical analysis to derive the probability of misdetection of the blinking sequences, and analytically determined the number of different blinking sequences that can be detected as a function of their length. This can be used to calculate the total number of UAVs that can be identified by this system.
- Experimental validation: we implemented a prototype of the proposed vision-based mutual identification system for UAVs and tested it outdoors.

The paper is organized as follows. In section 2, we describe the system model. This includes the models for the UAV clock signal, for the optical identification system, and for the optical transmission channel. In section 3, we formalize the problem of the visual identification system for UAVs. In section 4, we describe the theoretical framework for the construction of the blinking sequences for the LEDs using the Non-Return-to-Zero (NRZ) optical modulation (a popular optical modulation scheme). In section 5, we show how, instead of the NRZ, the use of the Manchester optical modulation (another popular optical modulation scheme) changes the properties of the identification system. In section 6, we study the effects of the UAVs clock on the performance of the identification system. Section 7 describes the experiments performed. Conclusions are provided in section 8.

2 System Model

We consider a group of UAVs composed of J members. Each UAV is equipped with the UVDAR system, see Fig. 1, which is composed mainly of three modules, see Fig. 2: the optical transmitter, the optical receiver, and the clock signal generator. We now discuss the three modules, and the optical channel model.

2.1 Clock signal

The clock signal's falling (or rising) edges indicate the instants when the receiver's camera shoots and when the LEDs of the transmitter can change state. An ideal clock signal should be stable and the interval between falling (or rising) edges should always remain constant. Furthermore, the clock signal frequency of the different UAVs of
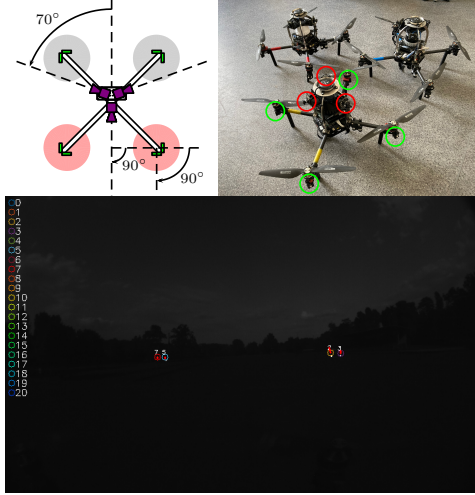
Springer Nature 2021 LaTeX template

or multi-UAV systems: theory and practice   3

Fig. 1 On top UAV platforms used in the experimental data acquisition. The frame is based on the *Holybro X500* platform, with an arm length from the center of 0.245 cm. Each unit is equipped with the UVDAR system, with three UV cameras and four pairs of UV LEDs. Each of the LED pairs is placed at the end of each arm, and cameras are attached as shown on the diagram to cover the entire horizontal surroundings. The LEDs are rated at 1 W input power, but we are driving them at 600 mW, producing cca 276 mW of radiometric power. On the bottom, view of the left UVDAR camera of UAV-0 in the outdoor flight experiment. The markers are correctly labeled based on the retrieved signal.

the group should be exactly the same. Unfortunately, due to physical impairments, the clock signals are not perfectly stable. Even if the nominal frequencies of all the clocks are the same, their true frequencies will differ slightly. These impairments and their effects on similar systems have been documented in the literature. For instance, in [15, 16], it was observed that the measured inter-frame interval of certain cameras is time-variant. In [9], it was noted in the context of smartphone cameras, that the nominal frame rate by software differs from the true frame rate and varies depending on the phone. As will be demonstrated in section 6, these irregularities on the clock signal limit the capacity of the optical identification system studied in this article. First, let us describe the model of the jth UAV clock signal, denoted by $c_j(t)$. Without loss of generality, we consider that the optical transmitter and receiver are controlled by the falling edges of $c_j(t)$. Based on the

mathematical models for clock signals described in [17, 18], we model the kth falling edge instant of $c_j(t)$ as:

$$t_{j,k} = T_j + n_{j,k} + t_{j,k-1}, \qquad k = 1, 2, \quad (2.1)$$

where $t_{j,0}$ is the instant when the system of the jth UAV is turned on; $t_{j,k}$ with $k \geq 1$ is the instant of the kth falling edge instant of $c_j(t)$; $T_j > 0$ is the true clock signal period of $c_j(t)$ and is modelled as a random variable with mean $\mathbb{E}[T_j] = T$, with T being the nominal period of the clock signal, with variance $\text{var}[T_j] = \sigma_T^2$. Due to the fabrication process uncertainties, different clocks will have slightly different oscillation frequencies, even if their nominal frequencies are the same. Thus, we consider that the set $\{T_j\}_{j=1}^J$ is composed of J statistically independent and identically distributed random variables. $n_{j,k}$ accounts for the frequency instability of the clock signal and is modelled as a white noise process.

2.2 Optical Transmitter

Fig. 2 (right) depicts the diagram of the optical transmitter for the quadrotor shown in Fig. 1. The transmitter is divided into M parallel branches. In this particular case, we select $M = 4$, i.e., one branch per UAV arm in Fig. 1. We can have more branches [19], but a discussion on such configurations is beyond the scope of this article. The transmitter modules are the following:
1) Binary stream generator: takes as inputs the clock signal $c_j^d(t)$ and a binary matrix $\mathbf{S}_j$ of size $M \times L$, which contains M binary sequences of length L. The mth binary stream generator produces the discrete-time stream $s_{j,m}$ composed of a continuously repeated concatenation of the binary sequence contained in the mth row of the matrix $\mathbf{S}_j$; $s_{j,m}[k]$ denotes the kth bit in the binary stream $s_{j,m}$. All the binary sequences used by the UAVs in the group are stored in a dictionary matrix $\mathbf{D}$; each row of the matrix $\mathbf{D}$ is a different binary sequence, and its identification number is the row number in which it is stored. The dictionary $\mathbf{D}$ is shared by all the UAVs in the group. Finally, the identification number embedded in the stream $s_{j,m}$ is the same as the identification number of the binary sequence used to generate it. **2) Encoder and Modulator (Enc./Mod.)**: the encoder codifies the binary discrete-time stream

Springer Nature 2021 LaTeX template
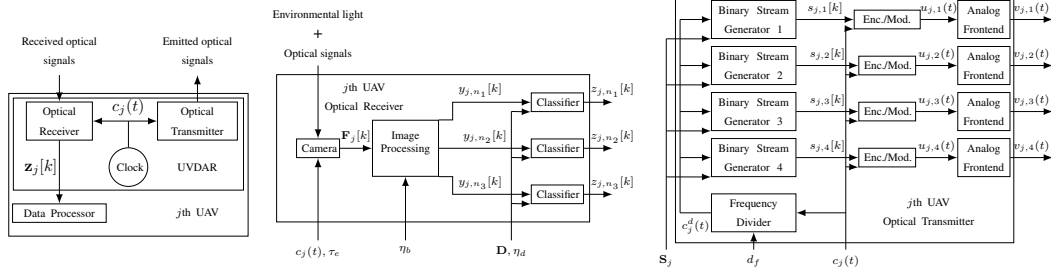
tion for multi-UAV systems: theory and practice

Fig. 2 UVDAR block diagram of the jth UAV (left). Optical receiver architecture for the jth UAV (center). Optical transmitter architecture for the jth UAV (right).

$s_{j,m}$ with a line code, such as NRZ or Manchester [20]. The modulator modulates the encoded signal with On-Off Keying (OOK) [21] to produce the continuous-time electrical signal $u_{j,m}(t) \in \{0, 1\}$. **3) Frequency divider**: divides the clock signal frequency by a factor d_f. The modulator and the encoder are both driven directly with the clock signal $c_j(t)$. But, the binary stream generator is driven by $c_j^d(t)$. If the Manchester code is used, then each bit consists of two minibits and the frequency of $c_j^d(t)$ must be half the frequency of $c_j(t)$, i.e., we need $d_f = 2$. Alternatively, if the NRZ code is used, then each bit consists of only one mini-bit. Thus, the frequencies of $c_j^d(t)$ and $c_j(t)$ must be equal, i.e. $d_f = 1$. **4) Analog frontend**: this transforms the binary electrical signal $u_{j,m}(t)$ into the optical signal $v_{j,m}(t) = Pu_{j,m}(t)$, where P is the emitted optical power. Each frontend has two LEDs that emit the same optical signal with a wavelength of 395 nm. Both LEDs are mounted orthogonally (see Fig. 1) to increase the angular visibility range of the optical signal.

2.3 Optical Receiver

The architecture of the optical receiver of the jth UAV is shown in Fig. 2 and is composed of the following modules: **1) Camera**: a grey scale UV sensitive camera mounted on the UAV as shown in Fig. 1 (center). The camera is coupled with an optical filter that allows UV light to pass and filters-out most visible light, see [22]. The filter attenuates most of the background light and facilitates the detection of the UV light emitted by the other UAVs. The camera shoots at every falling edge of the clock signal $c_j(t)$ with an exposure time τ_e. The kth frame captured is denoted as

$\mathbf{F}_j[k]$. **2) Image Processing**: this module must detect the bright spots potentially generated by the UV LEDs from the other UAVs in the group, track their motion on the screen, and then extract the optical signal from their blinking patterns. To do this, the frame $\mathbf{F}_j[k]$ is first binarized with the threshold η_b to produce $\bar{\mathbf{F}}_j[k]$. This simplifies the distinction between the background and bright spots potentially generated by UV LEDs, see Fig. 1. When a new bright spot is detected in $\bar{\mathbf{F}}_j[k]$, a serialized service number n_s is assigned and the following operations take place simultaneously: i) the coordinates of the central pixel of the n_sth bright spot is estimated $\hat{\mathbf{p}}_{j,n_s}[k]$, and its onscreen motion begins to be tracked; ii) the pixel with coordinates $\hat{\mathbf{p}}_{j,n_s}[k]$ is read in $\bar{\mathbf{F}}_j[k]$, and the values are stored as a binary time series $y_{j,n_s}[k]$. The instant when the time series associated with the n_s bright spot is created is denoted as its *birth time* t_{b,j,n_s}; iii) a classifier instance is created to process the time series y_{j,n_s}. As long as the n_sth bright spot is successfully tracked, the associated time series y_{j,n_s} remains *alive*. But, once the tracking fails, the time series y_{j,n_s} *dies* and the associated classifier instance is destroyed. We denote this instant as the *death time* t_{d,j,n_s} of time series y_{j,n_s}. Possible reasons for tracking failure may include LED occlusions, fast movements of the bright spot on the camera frame, or LED blinking patterns with long times off. **3) Classifier**. Each classifier takes the dictionary $\mathbf{D}$ described in section 2.2 and the last L bits received in the binary stream y_{j,n_s} as input. The classifier output is the time-series $z_{j,n_s}[k]$, which is then fed into the higher level modules. The first objective of the classifier is to determine if $\{y_{j,n_s}[m]\}_{m=k-L+1}^{k}$ was generated with a binary sequence contained

in the dictionary $\mathbf{D}$. This allows for discarding the bright spots generated by sources other than the UAVs. This is done by calculating the correlation of $\{y_{j,n_s}[m]\}_{m=k-L+1}^{k}$ with the sequences contained into the dictionary $\mathbf{D}$, and then comparing it with a detection threshold η_d. If the classifier decides that $\{y_{j,n_s}[m]\}_{m=k-L+1}^{k}$ was not generated by a binary sequence contained in the dictionary $\mathbf{D}$, then it produces $z_{j,n_s}[k]=-1$. In the contrary case, the classifier estimates the identification number of the binary stream y_{j,n_s}; $z_{j,n_s}[k]$ takes on this number.

2.4 Optical channel model

We reasonably assume that the exposure time τ_e is smaller than the coherence time of the background illumination signal. Then, regarding the optical channel between an LED from the jth UAV and the camera from the ℓth UAV, we have:

$$x_\ell[k] = h_\ell(t_k) \int_{t_{\ell,k}}^{t_{\ell,k}+\tau_e} v_j(t)\mathrm{d}t + n_\ell[k], \quad (2.2)$$

where $x_\ell[k]$ is the pixel value from the kth frame captured by the ℓth UAV camera, $h_\ell(t_k)$ is the optical channel gain, $v_j(t)$ is the optical power emitted by the LED of the jth UAV, see 2.2, and $n_\ell[k]$ is the noise generated at the pixel. In general, the integral representing the exposure process in (2.2) becomes [17]:

$$\int_{t_k}^{t_k+\tau_e} p(t)\mathrm{d}t = \tau_e P(a[k]s[k_t]+(1-a[k])s[k_t+1]),$$
$$(2.3)$$

where k_t is related to k by:

$$k_t = \left\{ \arg\min_n \{|\ t_{\ell,k}-t_{j,n}\ |\} : t_{j,n} \leq t_{\ell,k} < t_{j,n+1} \right\}.$$
$$(2.4)$$

During the exposure process, a bit transition may occur in $v_j(t)$. This is modelled by the random process $a[k] \in [0,1]$, whose behaviour depends on the relative uncertainties of the clock signals from the transmitter and receiver, as well as on the exposure time τ_e.

3 Problem description and proposed solution

The mutual identification system must determine the identification numbers associated with the optical signals emitted by the LEDs of the UAVs in the group, as in section 2.2. This identification system can be used to estimate the relative location and pose of the UAVs [19]. In this case, each branch of the optical transmitters of the UAVs will transmit different optical signals, and thus all rows in $\mathbf{S}_j$ will be different. On the other hand, the identification system can be used to estimate only the relative positions of the UAVs [23, 8, 24]. In this case, each branch of the optical transmitters will transmit the same optical signals, and therefore all the rows in $\mathbf{S}_j$ will be identical. The matrix $\mathbf{S}_j$ is related to the dictionary $\mathbf{D}$ by:

$$\mathbf{S}_j = \mathbf{A}_j\mathbf{D}, \quad (3.1)$$

where $\mathbf{A}_j$ is an $M \times N$ binary matrix that we call the assignation matrix. It is a design parameter to select the binary sequences used by the jth UAV, and also to determine in which branches they will be emitted.

We seek to design the dictionary matrix $\mathbf{D}$ for the system described in section 2 to minimize the expected identification time for a fixed number of different optical signals, i.e. minimize the expected identification time given for a fixed number of rows of $\mathbf{D}$. We define the identification time of an optical signal as the time elapsed from its *birth time* (defined in section 2.3) until the time when the classifier assigned to the signal successfully determines its identification number. The design of the ssignation matrices $\{\mathbf{A}_j\}_{j=1}^{J}$ is beyond the scope of this article. Thus, they will be considered fixed with an arbitrary configuration.

Regarding the encoder, we select the NRZ coding as it maximizes the bit rate for the OOK modulation (as long as synchronization problems are not considered) [25]. In section 5, we briefly discuss utilization of the Manchester coding.

tion for multi-UAV systems: theory and practice

4 Binary Sequences Construction and Combinatorial Analysis

The performance of the mutual vision-based identification system discussed in this article strongly depends on the set of binary sequences contained in the dictionary **D**, as mentioned in section 2.3. Let $\mathcal{X}^L$ be the set of all the binary sequences in the dictionary matrix **D** with dimensions $M \times L$. Let $\mathbf{b}_n$ denote the nth binary sequence in $\mathcal{X}^L$ and $\mathbf{b}_n[k]$ denote its kth bit, where $k = 0, 1, \cdots, L-1$. For simplicity, we disregard the effects of the clock signals mismatches in this section, but they will be studied theoretically in section 6 and experimentally in section 7.

Before we proceed, we describe the requirements of the identification system receiver of Fig. 2 (center). The image processing module in Fig. 2 (center) must reliably detect the bright spots generated by the LEDs of the group's UAVs. It must also discriminate bright spots generated by the UAVs from the bright spots generated by random environmental lights. As the UAVs move and their relative positions change, the image processing module must track the motion of the blinking lights emitted by the LEDs of the UAVs. The classifier must determine the true identification number of the analyzed optical sequences as fast as possible. In addition, the identification system must support as many different optical signals as possible to enable its use for a large group of of UAVs.

The requirements described above dictate the following requirements for the binary sequences in the set $\mathcal{X}^L$:

1) To facilitate the detection and tracking of the bright spots generated by the LEDs, we ensure a minimum average power of the emitted optical signals. Since we are using the OOK modulation with the NRZ coding, the average power of the optical signal associated with the binary sequence $\mathbf{b}_n$ is proportional to the average power of the binary sequence. To ensure a minimum average power on all emitted optical sequences, we constrain all of the binary sequences to satisfy:

$$\|\mathbf{b}_n\|_0 \geq \bar{b}L, \qquad (4.1)$$

where $\bar{b} \in [0, 1]$ is the desired normalized minimum average power, and $\|.\|_0$ is the L_0-norm.

2) Many bright spots on the camera frames that are not generated by UAV LEDs are sunlight reflections. Some of these reflections are generated by static reflectors and appear on many consecutive camera frames as a constant bright spots. We help to discriminate valid binary sequences from these reflections by limiting the maximum time that any LED can be continuously turned on. Thus, we limit to N_1 the number of circularly consecutive bits with value '1' for each binary sequence $\mathbf{b}_n$.

3) The image processing module must track the motion of all bright spots detected on the camera frame. One way to implement this tracker is by using the Hough transform [26] as in [8]. But, regardless of the particular implementation, the general behaviour of the tracker is as follows. When the bright spot is detected on the camera frame, the tracker locks to the central pixel of the bright spot and starts tracking it. Since the UAV LEDs are blinking, when the LED is turned off, the tracker must predict the central pixel of the bright spot, which should appear once the LED is turned on again. The longer the LED remains off, the larger the uncertainty of the central pixel location. If this uncertainty grows too large, the tracking will fail. To reduce the tracking failure, we limit the time that each LED can remain turned off by restricting $\mathbf{b}_n$ to have no more than N_0 circularly consecutive bits with value '0'.

4) The emitted optical signals are periodic with a period of L bits. The L most recent bits received at the input of the classifier at time instant k, assuming no bit errors, are:

$$\begin{bmatrix} y[k] \\ y[k-1] \\ \vdots \\ y[k-L+2] \\ y[k-L+1] \end{bmatrix} = \begin{bmatrix} \mathbf{b}_n[\mathrm{mod}(L-1+d, L)] \\ \mathbf{b}_n[\mathrm{mod}(L-2+d, L)] \\ \vdots \\ \mathbf{b}_n[\mathrm{mod}(d+1, L)] \\ \mathbf{b}_n[\mathrm{mod}(d, L)] \end{bmatrix}, \qquad (4.2)$$

where d is a random variable uniformly distributed within the discrete set $\{0, 1, \ldots, L-1\}$, representing the lack of time synchronization between the optical receiver and the optical transmitter. The classifier must identify $\mathbf{b}_n$, regardless of the random shift d and without its knowledge. Thus, any two binary sequences $\mathbf{b}_n$ and $\mathbf{b}_m$ are considered

Springer Nature 2021 LaTeX template

or multi-UAV systems: theory and practice 7

equal if one is a circularly shifted version of the other, in which case we say that they are circularly equivalent.

5) When the Signal-to-Noise Ratio (SNR) is poor, the raw Bit Error Rate (BER) is large, and can result in long identification times and constant identification failures. To alleviate this, we can add some robustness by increasing the circular Hamming distance of the set $\mathcal{X}^L$, defined as:

$$D(\mathcal{X}^{\mathcal{L}}) = \min_{\mathbf{b}_n, \mathbf{b}_m \in \mathcal{X}^{\mathcal{L}}} H_c(\mathbf{b}_n, \mathbf{b}_m). \qquad (4.3)$$

where $H_c(\mathbf{b}_n, \mathbf{b}_m) \triangleq \min_d \|\mathbf{b}_n \oplus c(\mathbf{b}_m, d)\|_0$ is the circular Hamming distance between the binary sequences $\mathbf{b}_n$ and $\mathbf{b}_m$; $c(\mathbf{b}_n, d)$ is the binary sequence $\mathbf{b}_n$ after being circularly shifted d bits to the right; and $\oplus$ is the XOR logic operator. If $D(\mathcal{X}^L) = 1$, then any single bit error can transform a valid binary sequence into another valid binary sequence. Thus, it is impossible to determine if the binary sequence was correctly decoded. If $D(\mathcal{X}^L) = 2$, any single bit error will transform a valid binary sequence into an invalid binary sequence. Thus, it becomes possible to detect single bit errors, but it will not be possible to correct them. If $D(\mathcal{X}^L) = 3$, then any single bit error will transform a valid binary sequence into an invalid binary sequence. However, the circular Hamming distance of this erroneous invalid binary sequence to the original binary sequence will be shorter than to any other valid binary sequence. Thus, it will be possible to detect and correct single bit errors.

4.1 Binary sequence set generation and analysis

After establishing the sequence requirements, we construct $\mathcal{X}^L$ and study its cardinality. To do this, we use the algorithm 1 with the following inputs: the set $\mathcal{S}^L$ of all the 2^L binary sequences of length L, the minimum value allowed for each sequence average $\bar{b}$ (see (4.1)), the maximum number N_1 (N_0) of circularly consecutive bits with value '1' ('0') for each sequence, and the circular Hamming distance H_m for $\mathcal{X}^L$.

We now discuss each step of algorithm 1 and calculate the cardinality of the output set $\mathcal{X}^L$. To do this, we partition $\mathcal{S}^L$ into $L + 1$ partitions, $\{\mathcal{S}_\ell^L\}_{\ell=0}^L$, where $\mathcal{S}_\ell^L$ is the partition containing all binary sequences $\mathbf{b} \in \mathcal{S}^L$ that satisfy $\|\mathbf{b}\|_0 = \ell$.

Algorithm 1 Sequences generation for NRZ coding

1: **procedure** $\mathcal{X}^L = f(\mathcal{S}^L, \bar{b}, N_1, N_0, H_m)$
2:     $\mathcal{A}^L$=PowerTest($\mathcal{S}^L, \bar{b}$)
3:     $\mathcal{B}^L$=CircularityTest($\mathcal{A}^L$)
4:     $\mathcal{C}^L$=OnesTest($\mathcal{B}^L, N_1$)
5:     $\mathcal{D}^L$=ZerosTest($\mathcal{C}^L, N_0$)
6:     $\mathcal{E}^L$=HammingTest($\mathcal{D}^L, H_m$)
7:     **return** $\mathcal{E}^L$
8: **end procedure**

The same partition is applied to each set. The cardinality of $\mathcal{S}_\ell^L$ is given by the binomial coefficient L choose ℓ:

$$\mid \mathcal{S}_\ell^L \mid = \begin{pmatrix} L \\ \ell \end{pmatrix}. \qquad (4.4)$$

4.1.1 Power test

The power test sets the minimum power of the emitted optical signals to $\bar{b}P$. This is done by discarding the subsets $\mathcal{S}_\ell^L$ with $\ell < L\bar{b}$. Thus, the cardinality of $\mathcal{A}^L$ is:

$$\mid \mathcal{A}^L \mid = \sum_{\ell=\lceil L\bar{b} \rceil}^{L} \mid \mathcal{S}_\ell^L \mid. \qquad (4.5)$$

4.1.2 Circularity test

This test ensures that all binary sequences in $\mathcal{B}^L$ are circularly different. To do this, we extract a sequence $\mathbf{b}$ from $\mathcal{A}_\ell^L$, include it into $\mathcal{B}_\ell^L$, and eliminate from[1] $\mathcal{A}_\ell^L$ all of the sequences that are circularly equivalent to $\mathbf{b}$. We repeat this for each sequence in $\mathcal{A}_\ell^L$ until $\mid \mathcal{A}_\ell^L \mid = 0$. Then, we repeat this process for all the remaining partitions of $\mathcal{A}^L$. Each binary sequence $\mathbf{b} \in \mathcal{A}_\ell^L$ has, at most, $L - 1$ circularly equivalent sequences[2] in $\mathcal{A}_\ell^L$. Therefore, for $0 < \ell < L$, we can approximate the cardinality of $\mathcal{B}_\ell^L$ as:

$$\mid \mathcal{B}_\ell^L \mid \approx \mid \mathcal{A}_\ell^L \mid / L, \qquad (4.6)$$

while $\mid \mathcal{B}_0^L \mid = 1$ and $\mid \mathcal{B}_L^L \mid = 1$.

4.1.3 Ones and zeros tests

These tests ensure that each of the binary sequences has no more than N_1 circularly consecutive bits with value '1', and no more than

---

[1] All the circularly equivalent sequences to $\mathbf{b} \in \mathcal{A}_\ell^L$ have the same L_0 norm, and thus belong to the same partition.
[2] It can have less if the sequence presents some symmetries.

tion for multi-UAV systems: theory and practice

N_0 circularly consecutive bits with value '0'. Let us start with the *Ones test*. All sequences within partitions $\{\mathcal{B}_\ell^L\}_{\ell \leq N_1}$ have no more than N_1 circularly consecutive bits with value '1', since their L_0 norm is not larger than N_1 by definition. Consequently, $\mid \mathcal{C}_\ell^L \mid = \mid \mathcal{B}_\ell^L \mid$ for $\ell \leq N_1$. But, partitions $\{\mathcal{B}_\ell^L\}_{\ell > N_1}$ have sequences with more than N_1 circularly consecutive bits with value '1' and must be eliminated. To calculate $\{\mid \mathcal{C}_\ell^L \mid\}_{\ell \in (N_1, L-1]}$, we proceed as follows. Due to the circular equivalence, every single sequence $\mathbf{b} \in \mathcal{B}_\ell^L$ with $N_1 < \ell < L$ having more than N_1 circularly consecutive bits with value '1' can be written, after some circular shifting, in the form of the following row vector:

$$\mathbf{b} = [1_{N_1+1}, v_{L-N_1-2}(\ell - N_1 - 1), 0], \quad (4.7)$$

where 1_x is a binary row vector of length x and $\|1_x\|_0 = x$, $v_x(y)$ is any binary row vector of length x with $\|v_x(y)\|_0 = y$. The number of all sequences $\mathbf{b} \in \mathcal{B}_\ell^L$ in (4.7) is determined by the number of different vectors $v_{L-N_1-2}(\ell - N_1 - 1)$, given by the binomial coefficient $L - N_1 - 2$ choose $\ell - N_1 - 1$:

$$\Delta_C(\ell) = \binom{L - N_1 - 2}{\ell - N_1 - 1}. \quad (4.8)$$

The number of sequences $\mathbf{b} \in \mathcal{B}_\ell^L$ that violate the constraint of the maximum allowed number of circularly consecutive bits with value '1' is approximately $\Delta_C(\ell)$. Thus, we have:

$$\mid \mathcal{C}_\ell^L \mid \approx \max\left(\mid \mathcal{B}_\ell^L \mid -\Delta_C(\ell), 0\right). \quad (4.9)$$

The *zeros test* is complementary to the *ones test* in algorithm 1. Thus, we use a similar procedure to estimate $\mid \mathcal{D}_\ell^L \mid$.

If $N_1 \geq L - N_0$, then the *zeros-test* acts only on partitions that were not modified by the *ones-test*. Thus, using the same method used to derive (4.8)-(4.9), we obtain for the *zero-test*:

$$\mid \mathcal{D}_\ell^L \mid \approx \max\left(\mid \mathcal{C}_\ell^L \mid -\Delta_D(\ell), 0\right), \quad (4.10)$$

$$\Delta_D(\ell) = \binom{L - N_0 - 2}{L - \ell - N_0 - 1}. \quad (4.11)$$

If $N_1 < L - N_0$, then we can divide the partitions into three groups: i) partitions affected by the *zeros-test* only, i.e., partitions that satisfy $\mathcal{C}_\ell^L = \mathcal{B}_\ell^L$ and $\mathcal{D}_\ell^L \neq \mathcal{C}_\ell^L$. These partitions are given by $\ell \in 0, 1, \cdots, N_1 - 1$; ii) partitions

affected by both tests, i.e., partitions that satisfy $\mathcal{C}_\ell^L \neq \mathcal{B}_\ell^L$ and $\mathcal{D}_\ell^L \neq \mathcal{C}_\ell^L$. These partitions are given by $\ell \in N_1, N_1 + 1, \cdots, L - N_0$; and iii) partitions affected only by the *ones-test*, i.e., partitions that satisfy $\mathcal{C}_\ell^L \neq \mathcal{B}_\ell^L$ and $\mathcal{D}_\ell^L = \mathcal{C}_\ell^L$. These partitions are given by $\ell \in L - N_0 + 1, L - N_0 + 2, \cdots, L$.

The cardinality of $\{\mathcal{D}_\ell^L\}_{\ell \in [0, N_1)}$ is calculated using (4.10)-(4.11). The cardinality of $\{\mathcal{D}_\ell^L\}_{\ell \in (L-N_0, L]}$ remains the same as that of the partitions $\{\mathcal{C}_\ell^L\}_{\ell \in (L-N_0, L]}$. Regarding $\{\mathcal{D}_\ell^L\}_{\ell \in [N_1, L-N_0]}$, (4.10)-(4.11) provide a poor cardinality estimation, as they disregard that some sequences that violate the *zeros-test* also violate the *ones-test* and thus were already discarded. After extensive numerical analysis, we derived the following heuristic approximation for the cardinality of partitions $\{\mathcal{D}_\ell^L\}_{\ell \in (L-N_0, L]}$:

$$\mid \mathcal{D}_\ell^L \mid \approx \max(\mid \mathcal{C}_\ell^L \mid -\Delta'_D(\ell), 0), \quad (4.12)$$

$$\Delta'_D(\ell) = \max(\Delta_D(\ell) - \Delta_C(\ell), 0), \quad (4.13)$$

where $\Delta'_D(\ell)$ is based on the difference between the number of sequences eliminated by the *one-test* and those eliminated by the *zero-test*. A good cardinality approximation for every partition $\mathcal{D}_\ell^L$ can be obtained by combining equations (4.5), (4.6), (4.8), (4.9), (4.11), (4.12), and (4.14). We show the cardinality of $\mathcal{D}^L$ and its estimation, using the above-mentioned equations, in the fifth and sixth columns of Table 1, respectively.

4.1.4 Hamming distance test

This test tries to maximize the cardinality of $\mathcal{X}^L$, while satisfying the circular Hamming distance $D(\mathcal{X}^L) = H_m$:

$$\underset{\mathbf{f}}{\text{maximize}} \quad \|\mathbf{f}\|_0 \quad (4.14)$$

$$\text{s.t.}$$
$$H_m[\mathbf{f}]_k[\mathbf{f}]_j \leq H_c(\mathbf{b}_k, \mathbf{b}_j),$$
$$j, k = 1, ..., \mid \mathcal{D}^L \mid, \quad j \neq k$$

where $\mathbf{b}_k$ is the kth sequence in $\mathcal{D}^L$, and $\mathbf{f}$ is a binary vector of length $\mid \mathcal{D}^L \mid$ that indicates which sequences are included in $\mathcal{X}^L$. If $\mathbf{b}_k \in \mathcal{X}^L$, then $[\mathbf{f}]_k = 1$. But, if $\mathbf{b}_k \notin \mathcal{X}^L$, then $[\mathbf{f}]_k = 0$. (4.14) describes a discrete combinatorial optimization problem. Verifying if a particular binary vector $\mathbf{f}^*$ constitutes an optimum solution requires the exploration of the full search space (composed

of $2^{|\mathcal{D}^L|}$ elements). Thus, as $\mid \mathcal{D}^L \mid$ grows, the problem becomes more computationally expensive to solve. Consequently, less computationally demanding suboptimal solutions are of interest.

For $H_m = 1$, we have that $\mathcal{X}^L = \mathcal{D}^L$. For $H_m = 2$, we can derive a suboptimal solution by considering the following three properties: i) if $\mathbf{b}_k, \mathbf{b}_j \in \mathcal{D}_\ell^L$, then $H_c(\mathbf{b}_k, \mathbf{b}_j) \geq 2$ because $\|\mathbf{b}_k\|_0 = \|\mathbf{b}_j\|_0$ and $H_c(\mathbf{b}_k, \mathbf{b}_j) > 0$. To transform any binary sequence $\mathbf{b}_k$ into any other sequence $\mathbf{b}_j$, at least two bit flips are needed: a 0 bit flip and a 1 bit flip. A single one bit flip on $\mathbf{b}_k$ would alter $\|\mathbf{b}_k\|_0$, and thus the resulting sequence would no longer belong to $\mathcal{D}_\ell^L$; ii) if $\mathbf{b}_j \in \mathcal{D}_\ell^L$ and $\mathbf{b}_k \in \mathcal{D}_{\ell \pm 2}^L$, then $H_c(\mathbf{b}_j, \mathbf{b}_k) \geq 2$ due to $\|\mathbf{b}_j\|_0 = \|\mathbf{b}_k\|_0 \pm 2$; and iii) if $\mathbf{b}_j \in \mathcal{D}_\ell^L$ and $\mathbf{b}_k \in \mathcal{D}_{\ell \pm 1}^L$, then $H_c(\mathbf{b}_j, \mathbf{b}_k) \geq 1$ due to $\|\mathbf{b}_j\|_0 = \|\mathbf{b}_k\|_0 \pm 1$. From these properties, we conclude that $D(\cap_k \mathcal{D}_{2k}^L) \geq 2$ and $D(\cap_k \mathcal{D}_{2k+1}^L) \geq 2$, where $\cap$ is the intersection operator. Thus, a suboptimal solution for $H_m = 2$ is:

$$\mathcal{X}^L = \begin{cases} \cap_k \mathcal{D}_{2k}^L & \text{if } \mid \cap_k \mathcal{D}_{2k}^L \mid > \mid \cap_k \mathcal{D}_{2k+1}^L \mid \\ \cap_k \mathcal{D}_{2k+1}^L & \text{otherwise} \end{cases} \tag{4.15}$$

and its cardinality is the sum of the cardinalities of the selected partitions, which we have already shown how to calculate.

Developing similar methods for $H_m > 2$ is extremely complicated due to the growing complexity of the relations among the binary sequences and the partitions. A suboptimal solution can be obtained using methods based on random search. It is simple to implement, but difficult to analyze.

It is possible to analytically derive a coarse cardinality estimation for the optimum solution when $H_m = 3$. To do this, we use a modified version of the definition of a sphere around a binary vector $\mathbf{c}$ as used in [27]:

$$\mathcal{S}_r(\mathbf{c}) \triangleq \{\mathbf{v} \in \mathcal{D}^L : H_c(\mathbf{c}, \mathbf{v}) \leq r\} \tag{4.16}$$

where $\mathbf{c}$ is the center of the sphere of radius r. If $\mathbf{b}_j, \mathbf{b}_k \in \mathcal{X}^L$ with $D(\mathcal{X}^L) = 3$, then $H_c(\mathbf{b}_j, \mathbf{b}_k) \geq 3$. The following properties also hold: i) the spheres of radius one of any two valid sequences do not overlap $\mathcal{S}_1(\mathbf{b}_j) \cap \mathcal{S}_1(\mathbf{b}_k) = \emptyset$; ii) the sphere of radius two of any valid sequences does not include any other valid sequence $\mathcal{S}_2(\mathbf{b}_j) \cap \mathbf{b}_k = \emptyset$; iii) the spheres of radius two (or the larger) of any two

valid sequences can overlap and so, in general, we have that $\mathcal{S}_n(\mathbf{b}_j) \cap \mathcal{S}_n(\mathbf{b}_k) \neq \emptyset$ for $n \geq 2$; and iv) regardless of the nature of $\mathcal{D}^L$ in (4.16), the *volume* of the sphere of radius one of any sequence is bounded as follows: $\mid \mathcal{S}_1(\mathbf{b}) \mid \leq L + 1$.

From the properties described above, we can think, in an oversimplified manner, of the optimization problem (4.14)-(4.15) as the problem of forming as many spheres of radius one, defined by (4.16), as possible while using the sequences in $\mathcal{D}^L$, where $\mathcal{X}^L$ is formed with the sequences that constitute the centers of all of the spheres. Following this reinterpretation of (4.14)-(4.15), a coarse estimation for the cardinality of $\mathcal{X}^L$ when $D(\mathcal{X}^L) = 3$ is the maximum number of spheres of radius one that can be formed with sequences in $\mathcal{D}^L$:

$$\mid \mathcal{X}^L \mid \approx \lceil \mid \mathcal{D}^L \mid /(L+1) \rceil . \tag{4.17}$$

In table 1, we plot, for a circular Hamming distance of 3, the cardinality of $\mathcal{X}^L$ in the seventh column and its estimate using (4.17) in the eighth column. For $L \leq 11$, equation (4.17) is accurate, but for larger values of L, it is poor.

Table 1 Cardinality Results

L	$\bar{b}$	N_1	N_0	$\mid \mathcal{D}^L \mid$	$\mid \hat{\mathcal{D}^L} \mid$	$\mid \mathcal{X}^L \mid$	$\mid \hat{\mathcal{X}^L} \mid$
8	0.1	6	4	29	28	4	4
8	0.2	6	6	32	31	5	4
8	0.5	3	7	14	13	2	2
10	0.3	7	3	72	70	8	7
10	0.4	3	6	56	54	6	6
10	0.5	7	2	42	41	4	4
11	0.2	4	9	148	148	11	13
11	0.2	4	3	97	98	9	9
11	0.2	6	8	172	172	11	15
12	0.1	6	7	326	321	20	26
12	0.4	3	7	159	153	13	13

5 Manchester coding

If we use Manchester coding instead of NRZ, the encoder/modulator must operate twice as fast as the binary stream generator and the frequency division factor must be $d_f = 2$ (see diagram in Fig. 2). In this case, each bit has a duration T_b of two periods of the clock signal $c_j(t)$, i.e., $T_b = 2T$. With Manchester coding, the LED always changes state in the middle of every bit.

ation for multi-UAV systems: theory and practice

Consequently, regardless of the binary stream signal $s_{j,m}$, the average power of the emitted optical signal is $\mathbb{E}[v_{jm}(t)] = 0.5P$. The LEDs will continuously be turned on for $2T$ at most, and also continuously turned off for $2T$ at most. Thus, the Manchester coding automatically satisfies some of the requirements listed in section 4. Thus, if we use Manchester coding, we can drop some lines from Algorithm 1 and use Algorithm 2 instead.

Algorithm 2 Sequences generation for Manchester coding

1: **procedure** $\mathcal{X}^L = f_M(\mathcal{S}^L, H_m)$
2:    $\mathcal{A}^L$=CircularityTest($\mathcal{S}^L$)
3:    $\mathcal{B}^L$=HammingTest($\mathcal{A}^L, H_m$)
4:    **return** $\mathcal{B}^L$
5: **end procedure**

Algorithm 2 is less restrictive than Algorithm 1. It discards less sequences and, under the same conditions, generates a set $\mathcal{X}^L$ with higher cardinality. In other words, we require shorter sequence lengths L to obtain the set $\mathcal{X}^L$ with a desired number of sequences. However, the bit duration T_b when using the Manchester code is twice that of the NRZ code. To compare both codes fairly, we note from Algorithm 1 and Algorithm 2 that $\mid f_M(\mathcal{S}^L, H_m) \mid = \mid f(\mathcal{S}^L, 0.5, 2, 2, H_m) \mid$. Using this equivalence, we generate various sets of binary sequences with Algorithms 1 and 2 for comparison. The result is shown in table 2. The left part of the table shows the results obtained by using Algorithm 1 with the NRZ coding, where N_{NRZ} is the number of obtained sequences and LT_b/T is the normalized sequence duration. On the right side of the table, we observe the same information for the sequences obtained using algorithm 2 with the Manchester coding. We note that, for a given value of N_{NRZ}, the sequence duration is shorter. Thus, the NRZ coding results in shorter sequence durations and consequently shorter identification times.

Lastly, the use of optical signals of different frequencies as is done in [19, 24] is extremely inefficient. Using the Fast Fourier Transform (FFT) we can demonstrate that with this strategy we can only produce $L/2$ different sequences of length L.

Table 2 NRZ and Manchester Comparison

N_{NRZ}	L	H_m	LT_b/T	N_{Man}	L	H_m	LT_b/T
6	8	1	8	6	5	1	10
11	10	1	10	12	6	1	12
24	12	1	12	34	8	1	16
7	14	3	14	8	10	3	20
16	16	3	16	18	12	3	24
28	18	3	18	29	13	3	26

6 Identification Time Analysis

Next, we study the identification time of the sequences considering the clock signal impairments. We focus on the link from one LED of UAV-1 to the camera of UAV-0. For simplicity, we assume an errorless detection and perfect tracking of the bright spot generated by the LED.

6.1 Ideal clock signals

When all clock signals are stable (i.e., $n_{j,k} = 0$ in (2.1)) and have the same true period (i.e., $T_j = T$), then (2.1) becomes:

$$t_{j,k_j} = k_j T + t_{j,0}, \qquad (6.1)$$

where k_j is the local discrete-time index of the jth UAV. From (6.1), it is clear that the clock signals of the optical transmitter of the UAV-1 and of the optical receiver of the UAV-0 operate at exactly the same rate. Thus, the receiver always takes one sample per bit transmitted, and the only source of bit error detection comes from the noise discussed in section 2.4.

When $H_m = 1$, the classifier must accumulate L consecutive errorless bits to identify the binary sequence. In this case, the minimum detection time (normalized over T) is L, which occurs when the first L bits received are errorless. Given a bit error probability p_b, the following identification time T_d probabilities hold:

$$Pr(T_d < L) = 0, \qquad (6.2)$$
$$Pr(T_d = L) = (1 - p_b)^L. \qquad (6.3)$$

For $Pr(T_d = L + m)$ with $m = 1, \cdots, L$, the most recent L bits received must be errorless and the $m - L$ bit (counted backwards) must be erroneous. Therefore, we have:

$$Pr(T_d = L + m) = p_b(1 - p_b)^L. \qquad (6.4)$$

or multi-UAV systems: theory and practice 11

For $m \in 1, 2, \cdots, L-1]$, we have

$$Pr(T_d = 2L + m) = 1 - p_b(1 - p_b)^{2L} - (m-1)p_b^2(1-p_b)^{2L}, \tag{6.5}$$

and for $m \geq L$, we have

$$Pr(T_d = 2L + m) = \left(1 - \sum_{k=0}^{m-1} Pr(T_d = L + k)\right) \times p_b(1 - p_b)^L. \tag{6.6}$$

From (6.4)-(6.6), the expected identification time is:

$$\mathbb{E}[T_d] = \left[\frac{3p_b}{2}L^2 + \left(1 + \frac{p_b}{2}\right)L\right](1 - p_b)^L$$
$$+ \left[\sum_{n=2L+1}^{\infty} n\left(1 - \sum_{k=L}^{n-L-1} Pr(T_d = k)\right)\right]$$
$$\times (1 - p_b)^L$$

From (6.7), we generally observe that the expected identification time is a strictly increasing non-linear function of the sequence length L. This analysis also holds for $H_m = 2$, but for $H_m = 3$, the analysis is more complex.

6.2 Stable clock signals with uncertain oscillation frequency

A more realistic case is when all clock signals are stable, but we consider the uncertainty due to the fabrication process in the clock period (i.e., $\sigma_T^2 \neq 0$). Then, (2.1) becomes:

$$t_{j,k_j} = k_j T_j + t_{j,0}. \tag{6.7}$$

After doing some algebra in (6.7) and (2.4), k_t becomes:

$$k_t = \left\lfloor \frac{\Delta + k_0 T_0}{T_1} \right\rfloor \tag{6.8}$$
$$- \left\lceil \frac{2(t_{0,0} - t_{1,0})}{T_1 + T_0 + \text{sign}(t_{0,0} - t_{1,0})(T_1 - T_0)} \right\rceil,$$

where

$$\Delta = t_{0,0} - t_{1,0} \tag{6.9}$$

$$- \left\lfloor \frac{2(t_{0,0} - t_{1,0})}{T_1 + T_0 + \text{sign}(t_{0,0} - t_{1,0})(T_1 - T_0)} \right\rfloor.$$

The index k_t, defined in (2.4), is the value of the local discrete time index k_1 at the transmitter when the local discrete time index at the receiver is k_0. In other words, at local discrete time k_0, the receiver samples the k_tth bit emitted by the receiver.

Let us define the following random variable:

$$\delta_{0,1} \triangleq T_0/T_1 - 1. \tag{6.10}$$

As mentioned in section 2.1, T_{j_0} and T_{j_1} are statistically independent. We also assume that[3] $\sigma_T^2/T^2 \ll 1$, and that the skewedness of T_j is zero, i.e., its probability distribution is symmetric w.r.t. its mean. Using the Taylor series approximations in (6.10), we demonstrate that $\mathbb{E}[\delta_{0,1}] \approx 0$ and that:

$$\text{var}[\delta_{0,1}] = \sigma_\delta^2 \approx 2\sigma_T^2/T^2. \tag{6.11}$$

The r.h.s. of (6.8) is composed of a time-variant term and of a constant term. We rewrite the first term using (6.10):

$$\lfloor(\Delta + k_0 T_0)/T_1\rfloor = k_0 + \lfloor \Delta/T_1 + k_0\delta_{0,1}\rfloor. \tag{6.12}$$

We observe in (6.12) that the time variant term of k_t in (6.8) is composed of a linear term k_0 and a nonlinear function of $\delta_{0,1}$ and k_0. When the receiver's clock is slower than the transmitter's clock, we have $\delta_{0,1} > 0$, and the nonlinear function in (6.12) increases one unit approximately every $\lfloor 1/\delta_{0,1}\rfloor$ sampling instants. However, when the receiver's clock is faster than the transmitter's clock, we have $\delta_{0,1} < 0$, and the nonlinear function in (6.12) decreases one unit approximately every $\lfloor 1/ \mid \delta_{0,1} \mid\rfloor$ sampling instants. Consequently, every $\approx \lfloor 1/ \mid \delta_{0,1} \mid\rfloor$ sampling instants, the receiver will miss a bit (if $\delta_{0,1} < 0$) or duplicate a bit (if $\delta_{0,1} > 0$). If the length of the emitted binary sequence, L, is larger than $\lfloor 1/ \mid \delta_{0,1} \mid\rfloor$, then the sequence will always be received with a missing or a duplicated bit. Thus, to ensure correct reception, the binary sequence length must satisfy $L < \lfloor 1/ \mid \delta_{0,1} \mid\rfloor$. Since δ_{01} is actually a random variable, the probability that the transmission of a binary sequence with length L

---

[3]This is a realistic assumption for oscillators of reasonable quality.

ation for multi-UAV systems: theory and practice

fails every time is $\Pr\left(L \geq \mid \delta_{01} \mid^{-1}\right)$. From (6.11) and the Chebyshev's inequality, we bound this probability as:

$$\Pr\left(L \geq \mid \delta_{01} \mid^{-1}\right) \leq \sigma_\delta^2 L^2 = 2\sigma_T^2 L^2 / T^2, \quad (6.13)$$

The probability that transmission through this link will work correctly is the complement to (6.13). The resulting inequality is the key to determining the maximum sequence length L for a group of J UAVs. From (6.13), the probability that all possible $J(J-1)$ links can operate correctly is:

$$p_g\left(J\right) = \Pr\left(\delta_J < L^{-1}\right) \qquad (6.14)$$

The random variable δ_J is the maximum value of the $J(J-1)$ identically distributed random variables $\{\mid \delta_{j,k} \mid\}_{j \neq k}$. However, that set of $J(J-1)$ variables is generated by only J independent random variables $(\{T_j\}_{j=1}^J)$. Thus, the random variables $\{\mid \delta_{j,k} \mid\}_{j \neq k}$ are not statistically independent. But, if we neglect this, then we can make the following approximation:

$$p_g\left(J\right) \approx \left(\Pr\left(\mid \delta_{jk} \mid < L^{-1}\right)\right)^{J(J-1)}, \qquad (6.15)$$

Further, using (6.13), we obtain:

$$p_g\left(J\right) \geq \left(1 - 2\sigma_T^2 L^2 / T^2\right)^{J(J-1)}. \qquad (6.16)$$

The probability $p_g\left(J\right)$ that all optical links operate correctly in the group of J UAVs is lower bounded according to (6.16). Consequently, if we want all the optical links in a group of J UAVs to work correctly with a probability that is lower bounded by $p_g\left(J\right)$, then the sequence length must satisfy:

$$L_{max} \triangleq \frac{T}{\sigma_T} \sqrt{\frac{1}{2}\left(1 - \exp\left(\frac{\ln\left(p_g\left(J\right)\right)}{J(J-1)}\right)\right)} \geq L. \tag{6.17}$$

The maximum sequence length L_{max} is proportional to the nominal clock period T and inversely proportional to the standard deviation of the clock period σ_T. L_{max} is a decreasing function of the number of UAVs within the group.

7 Simulations and Experiments

7.1 Hamming distance effect on the identification time

We consider a group of $J = 11$ UAVs with two different blinking sequences assigned to each UAV, i.e. $\dim(\mathbf{S}_j) = 2$. Thus, we need to construct a dictionary $\mathbf{D}$ with at least 22 different binary sequences. In addition, we set $\bar{b} = 0.4$, $N_1 = 7$, and $N_0 = 7$. Next, we consider two different cases. In case A, we construct a set with $H_m = 1$, and sequence length $L = 8$ (the minimum length that satisfies the desired number of sequences) which contains a total of 22 different sequences. In case B, we construct a second set with $H_m = 3$, and sequence length $L = 13$ (the minimum length that satisfies the desired number of sequences) which contains a total of 22 different sequences.

For both cases, we perform simulations to calculate the classification probability error p_{ce} and the expected identification time T_d for different bit probability errors p_b. We perform these simulations assuming $\sigma_T = 0$, i.e., perfect clock signals. The results are presented at the top of Table 3. For $p_b \leq 0.2$, the identification time T_d for case B is longer, but its classification error probability is lower. In the absence of a clock signal mismatch, the benefits of the reduced classification error probability provided by the robustness obtained by increasing the circular Hamming distance weaken as the bit probability error increases. This is because sets of sequences with a larger circular Hamming distance must have larger lengths to maintain their cardinality, thus presenting more errors.

Table 3 Simulation Results

$(\delta = 0), p_b$	$2 \cdot 10^{-1}$	10^{-1}	10^{-2}	10^{-3}
$\mathbb{E}[T_d]$ (case A)	21.404	12.751	8.369	8.031
$\mathbb{E}[T_d]$ (case B)	24.927	15.598	13.025	13.001
p_{ce} (case A)	0.789	0.538	0.073	0.0070
p_{ce} (case B)	0.687	0.322	0.006	0.0001
$(\mid \delta \mid = 0.01), p_b$	$2 \cdot 10^{-1}$	10^{-1}	10^{-2}	10^{-3}
$\mathbb{E}[T_d]$ (case A)	21.735	13.036	8.542	8.205
$\mathbb{E}[T_d]$ (case B)	25.025	16.134	13.267	13.205
p_{ce} (case A)	0.795	0.559	0.120	0.059
p_{ce} (case B)	0.704	0.368	0.064	0.057

7.2 Clock Period Uncertainty effect on the maximum identification capacity

Running the same simulations for the same set of binary sequences as was done in section 7.1, we now consider a mismatch between the transmitter and receiver clocks of $\delta = 0.01$ (i.e., one bit missed every 100 bits transmitted approximately) and present the results at the bottom of Table 3.

We observe a slight increase in the identification time, but the classification error probability has a more interesting behaviour: when the bit error probability is high, the classification error probability for case A gets close to that of case B, just slightly higher. When the bit error probability is medium, the classification error probability is significantly lower for case B. When the bit error probability is low, the classification error probability for both cases reaches a common lower limit, where the errors due to the clocks mismatch dominate the errors due to the individual bit decoding errors. Thus, increasing the SNR (which implies a further decrease of p_b) will not contribute to further reducing the classification error probability. When p_b is low, the classification error probability for case A is slightly lower than that for case B, because longer binary sequences are more affected by the clocks mismatch.
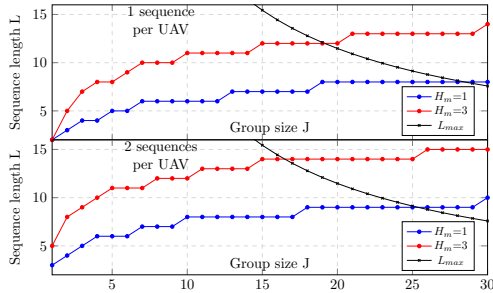

Fig. 3 In black, L_{max} for $p_g(J)$=0.999 and $T/\sigma_T = 10^4$. In blue and red, the minimum sequence length L of a set with circular Hamming distance 1 and Hamming distance 3, respectively, that satisfies the cardinality constraints for a group of J UAVs.

As mentioned in section 6.2, the variance σ_T^2 of the clock used in the UAVs determines the capacity of the identification system. To illustrate this, we analyse groups composed of J UAVs equipped

with reasonably accurate clocks ($T/\sigma_T = 10^4$). In Fig. 3, we plot in black the maximum sequence length L_{max} that can be used by the identification system, so that we have a probability $p_g(J)$ that all optical links can operate correctly (i.e., that any UAV can identify any other UAV). In blue, we plot the minimum sequence length L of a set with circular Hamming distance 1, required to assign one sequence per UAV (top) and two sequences per UAV (bottom); and in red, we plot the minimum sequence length L of a set with $H_m = 3$.

In the left image, we observe that for $L \geq 29$, both the blue and red curves are above L_{max}. This means that given the clocks used by the UAVs, it is not possible to assign one distinct sequence per UAV and ensure that any UAV in the group can identify any other UAV in the group with a probability of $p_g(J)$ or higher. In the right image where we assign two different sequences per UAV, this occurs for $L \geq 26$. On the left, for $20 \leq J \leq 28$ only, sets of sequences with $H_m = 1$ can ensure the proper operation of all optical links. For $J \leq 19$, we can use either sets of sequences with $H_m = 1$ or $H_m = 3$.

7.3 Camera interframe duration analysis

We recorded the camera interframe duration of the UAV shown in Fig. 1 during 400 seconds. The timings were recorded as those reported by the camera driver[4] used on the platforms. The nominal camera frame rate was set to 60 frames/s. The p.d.f. derived from the measurements histogram was recorded and compared to the Laplace distribution with its parameters estimated using maximum log-likelihood. We observed an excellent match between both distributions, and thus we conclude that the camera interframe duration follows a Laplace probability distribution.

To continue investigating the effect of the clock signal impairments, we perform the following indoor experiment. We place two UAVs (UAV-0 and UAV-1) of the same model as before 5 m apart on the floor of the laboratory. The UAV-0 camera operates with a nominal frame rate of 60 frames/s, and points to UAV-1. UAV-1 points one

---

[4]`https://github.com/ctu-mrs/bluefox2`

Springer Nature 2021 LaTeX template

ation for multi-UAV systems: theory and practice

arm towards the UAV-0 camera, while two adjacent arms remain oriented parallel to the UAV-0 camera image plane. We name these LEDs, from left to right on their camera image, as D0, D1, and D2.

The bit-rate of the blinking LEDs was $60.241\,Hz$ due to hardware limitations of the microcontroller architecture used in the LED driver. This creates a difference in the period of less than $66.7\,\mu s$ compared to the nominal camera period of $16.167\,ms$, with the exposure time of the camera set to $500\,\mu s$. The precision of the crystal clock of the LED driver incurs an additional error several orders of magnitude lower than the other sources of error. Each LED emits an optical signal generated with a different binary sequence. We generate a set of sequences $f(\mathcal{S}^7, 0.5, 3, 3, 1)$ using Algorithm 1. We then feed the binary stream generators (see Fig. 2 (right)) associated with LEDs D0, D1, and D2 the sequences 0010111, 0011011, and 0011101. Then, we record the UAV-0 camera footage for $180\,s$.

Due to the short distance between the UAVs, the pixels corresponding to the light emitted by the LEDs get almost saturated when the LEDs are turned on. Thus, the SNR at the receiver is high and the effects of the clock signal impairments and mismatch dominate over the effects of the noise.
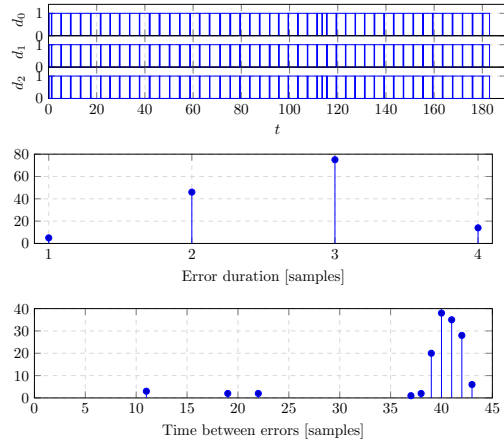

Fig. 4 The time progression and error analysis classification success signals for $f(\mathcal{S}^7, 0.5, 3, 3, 1)$ at $5\,m$.

In Fig. 4, we plot, for each LED, a binary signal that takes the value 1 when the binary sequence

embedded in the optical signal is correctly classified and 0 otherwise. In these plots, we note the presence of quasi periodic errors, which are generated by the mismatch between the transmitter and the receiver clock signals, as described in section 6. To analyze these errors in more detail, we plot the histograms of their duration and of the time between the errors measured from start-to-start in Fig. 4. From these histograms, we note two different types of errors: the first type of error is caused by noise at the receiver, and it has a duration of one single sample. The time between them does not follow any specific pattern; the second type of error is due to missing/duplicated bits caused by clock mismatches. In this experiment, they can last between two and four samples. The time between them is a random variable with a mean of around 40.6 samples and a variance of around 1.49 samples. The histograms in Fig. 4 show the following: i) if the optical signals are generated with sequences with a length of 39 bits or larger, they will always be received with errors; ii) the clock signal mismatch is time-variant, which is why the time between errors caused by missing/duplicated bits varies mostly between 39-42 samples; and iii) at high SNR, the main factor that limits the performance of the optical identification system is the mismatch between the transmitter and receiver clocks.

7.4 Dynamic outdoor testing

As proof of concept, we deployed a group of three UAVs (UAV-0, UAV-1, and UAV-2) equipped with the UVDAR system outdoors. We used binary sequences taken from the set generated with $f(\mathcal{S}^{14}, 3, 3, 0.5, 1)$ and assigned one sequence to each arm of each UAV, where each UAV emitted four unique optical signals through its LEDs. Fig. 1 shows a camera snapshot with correctly classified markers. The UAVs flew autonomously according to a formation enforcement technique developed in our laboratory. The ability to distinguish between the individual signals allowed the UAVs to identify each other and estimate their relative orientations. This flight allowed us to test the visual identification system in a more challenging and realistic scenario.

We recorded the content of the UAV-0 camera for $235\,s$ with a nominal frame rate of 60 frames/s. The clock signal of the optical transmitters of all

Springer Nature 2021 LATEX template

or multi-UAV systems: theory and practice 15

the UAVs operate with a nominal frequency of 60 Hz. The trajectories of the UAVs and the execution of the identification process are shown in a video of the experiment at the following link[5].

We evaluated the classification success signal for each individual LED of the UAV-1 and of the UAV-2 by the left camera of UAV-0. This signal takes on a value of one when the classification is successful and takes a zero value when there is an error in classification, or when the LED leaves the UAV-0 camera's Field of View (FoV). From the video of the experiment, we observe that usually only two LEDs per UAV are captured by the UAV-0 camera, although sometimes three LEDs can be captured simultaneously. We also note that, UAV-2 leaves the UAV-0 camera's FoV and is lost for some moments. The detection success is significantly more erratic than in the prior testing with the static transmitter and receiver, since the motion of the UAVs affects the optical signal retrieval. Additionally, the distances between the transmitters and receivers were at times greater, and the contrast of the active LEDs in the image was slightly lower due to sunlight. Despite this, the classification success was sufficient for the formation enforcement system to perform its function, which testifies to the practical applicability of the proposed identification system in real-world conditions.
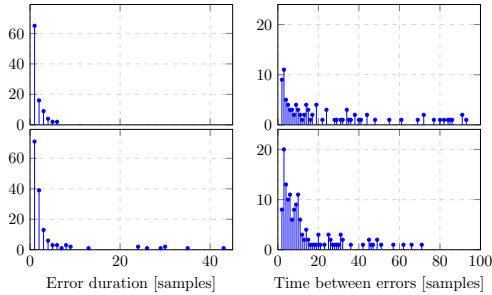

Fig. 5 Error duration and time between errors histograms for UAV-1 (top) and UAV-2 (bottom).

For both signals mentioned above, the histograms of the duration of the errors observed are plotted in Fig. 5. In Fig. 5, we plot the histograms of the time between the errors. The first thing to note in Fig. 5 is that most of the errors for both

UAVs last only a single sample. For UAV-1, single sample errors constitute 66.33% of the total errors. For UAV-2, single sample errors constitute 47.02% of the total errors. From Fig. 5, it can be observed that errors occur more often; this is due to the larger distances (lower SNR) and the additional effect of the blurring and tracking errors. Despite the challenging conditions, the probability of correctly detecting UAV-1 is 0.9311, and 0.6327 for detecting UAV-2. Further, the errors appear in short bursts as long as the Line of Sight (LoS) is present. This demonstrates that our identification system performs well in real scenarios.

8 Conclusion

In this paper, we studied the theoretical and practical aspects of UVDAR: a camera-based optical identification system for UAVs. Herein, it was shown how to optimize the optical signals emitted by the UAVs in order to maximize the number of detectable UAVs while minimizing identification time. Through theoretical analysis and experiments, we demonstrated that clock signal mismatches impose important limitations on the capacity of this visual identification system. This visual identification system was tested both indoors and outdoors, demonstrating successful operation with sufficient performance. The results of this work can be used to further optimize visual-based localization and identification systems, such as UVDAR, as well as to evaluate the capacity of this system as the base for an optical communication network for UAVs.

Declarations

---

[5]`http://mrs.felk.cvut.cz/uvdar-identification-sequences`

CHAPTER 2. ULTRAVIOLET RELATIVE LOCALIZATION SYSTEM *UVDAR*

References

[1] Ali, Q., *et al.*: A Review on Distributed Control of Cooperating Mini UAVS. International Journal of Artificial Intelligence & Applications **5**, 1–13 (2014)

[2] Vilca, J., Adouane, L., Mezouar, Y.: Adaptive Leader-Follower Formation in Cluttered Environment Using Dynamic Target Reconfiguration. In: Chong, N.-Y., Cho, Y.-J. (eds.) Distributed Autonomous Robotic Systems, pp. 237–254. Springer, Tokyo (2016)

[3] Chen, T., Gao, Q., Guo, M.: An improved multiple UAVs cooperative flight algorithm based on Leader Follower strategy. In: 2018 Chinese Control And Decision Conference (CCDC), pp. 165–169 (2018). https://doi.org/10.1109/CCDC.2018.8407124

[4] Mercado, D.A., Castro, R., Lozano, R.: Quadrotors flight formation control using a leader-follower approach. In: 2013 European Control Conference (ECC), pp. 3858–3863 (2013). https://doi.org/10.23919/ECC.2013.6669637

[5] et al., T.K.: A practical multirobot localization system. J. Intell. Robot. Syst. **76**, 539–562 (2014)

[6] et al., M.S.: System for deployment of groups of unmanned micro aerial vehicles in GPS-denied environments using onboard visual relative localization. Auton. Robots **41**(4), 919–944 (2017)

[7] Censi, A., *et al.*: Low-latency localization by active LED markers tracking using a dynamic vision sensor. In: 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 891–898 (2013). https://doi.org/10.1109/IROS.2013.6696456

[8] Walter, V., *et al.*: Mutual Localization of UAVs based on Blinking Ultraviolet Markers and 3D Time-Position Hough Transform. In: 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE), pp. 298–303 (2018). https://doi.org/10.1109/COASE.2018.8560384

[9] Le, N.T., Hossain, M., Jang, Y.M.: A survey of design and implementation for optical camera communication. Signal Processing: Image Communication **53**, 95–109 (2017). https://doi.org/10.1016/j.image.2017.02.001

[10] Saeed, N., *et al.*: Optical camera communications: Survey, use cases, challenges, and future trends. Physical Communication **37**, 100900 (2019). https://doi.org/10.1016/j.phycom.2019.100900

[11] Nguyen, T., Islam, A., Jang, Y.M.: Region-of-Interest Signaling Vehicular System Using Optical Camera Communications. IEEE Photonics Journal **9**(1), 1–20 (2017). https://doi.org/10.1109/JPHOT.2016.2644960

[12] Soares, M.R., *et al.*: Optical Camera Communications with Convolutional Neural Network for Vehicle-toVehicle Links. In: 2020 12th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP), pp. 1–6 (2020). https://doi.org/10.1109/CSNDSP49049.2020.9249499

[13] Takano, H., *et al.*: Visible Light Communication on LED-equipped Drone and Object-Detecting Camera for Post-Disaster Monitoring. In: 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), pp. 1–5 (2021). https://doi.org/10.1109/VTC2021-Spring51267.2021.9448902

[14] Takano, H., *et al.*: 300-Meter Long-Range Optical Camera Communication on RGB-LED-Equipped Drone and Object-Detecting Camera. IEEE Access **10**, 55073–55080 (2022). https://doi.org/10.1109/ACCESS.2022.3177140

[15] Nguyen, T., Le, N.T., Jang, Y.M.: Asynchronous scheme for unidirectional optical camera communications (OCC). In: 2014 Sixth International Conference on Ubiquitous and Future Networks (ICUFN), pp. 48–51 (2014). https://doi.org/10.1109/ICUFN.2014.6876747

[16] Hu, W., Gu, H., Pu, Q.: LightSync: Unsynchronized Visual Communication over Screen-Camera Links. In: Proceedings of the 19th Annual International Conference on Mobile Computing &; Networking. MobiCom '13, pp. 15–26. Association for Computing Machinery, New York, NY, USA (2013).

Springer Nature 2021 LATEX template

17

https://doi.org/10.1145/2500423.2500437. https://doi.org/10.1145/2500423.2500437

[17] Mao, W., Kahn, J.M.: Free-space heterochronous imaging reception of multiple optical signals. IEEE Transactions on Communications **52**(2), 269–279 (2004). https://doi.org/10.1109/TCOMM.2003.822755

[18] Messerschmitt, D.G.: Synchronization in digital system design. IEEE Journal on Selected Areas in Communications **8**(8), 1404–1419 (1990). https://doi.org/10.1109/49.62819

[19] Walter, V., *et al.*: UVDAR System for Visual Relative Localization With Application to Leader–Follower Formations of Multirotor UAVs. IEEE Robotics and Automation Letters **4**(3), 2637–2644 (2019). https://doi.org/10.1109/LRA.2019.2901683

[20] Chow, C.W., *et al.*: Mitigation of Optical Background Noise in Light-Emitting Diode (LED) Optical Wireless Communication Systems. IEEE Photonics Journal **5**(1), 7900307–7900307 (2013). https://doi.org/10.1109/JPHOT.2013.2238618

[21] Kahn, J.M., Barry, J.R.: Wireless infrared communications. Proceedings of the IEEE **85**(2), 265–298 (1997). https://doi.org/10.1109/5.554222

[22] BP365 Near-UV Bandpass Filter . https://midopt.com/filters/bp365/. Accessed: 2022-03-10

[23] Walter, V., Saska, M., Franchi, A.: Fast Mutual Relative Localization of UAVs using Ultraviolet LED Markers. In: 2018 International Conference on Unmanned Aircraft Systems (ICUAS), pp. 1217–1226 (2018). https://doi.org/10.1109/ICUAS.2018.8453331

[24] Petráček, P., Walter, V., Báča, T., Saska, M.: Bio-inspired compact swarms of unmanned aerial vehicles without communication and external localization. Bioinspiration & Biomimetics **16**(2), 026009. https://doi.org/10.1088/1748-3190/abc6b3

[25] Kwon, J.K.: Inverse Source Coding for Dimming in Visible Light Communications Using NRZ-OOK on Reliable Links. IEEE Photonics Technology Letters **22**(19), 1455–1457 (2010). https://doi.org/10.1109/LPT.2010.2062498

[26] Hart, P.E.: How the Hough transform was invented [DSP History]. IEEE Signal Processing Magazine **26**(6), 18–22 (2009). https://doi.org/10.1109/MSP.2009.934181

[27] Pless, V.: Introduction to the Theory of Error-Correcting Codes. John Wiley & Sons, Ltd, ??? (1998)

Chapter 3

In-flight visual relative localization and stabilization techniques

The task of detecting neighboring UAVs from a camera image builds upon the more general task of detecting and relatively localizing objects from cameras onboard UAVs. In real-world scenarios, relative localization based on computer vision is significantly more complex than in laboratory or industrial environments, with strictly defined targets and stable, predicable viewpoints. In addition, when aerial view of a scene is concerned, the motion and orientation of the observer at a given time are in practice not known exactly. If VRL is used in a feedback loop with a control that changes the viewpoint over time, the resulting action can, in turn, perturb the performance of the computer vision. This problem became significant during our participation in the MBZIRC 2020 competition. The competition, taking place in Abu Dhabi in February 2020, consisted of three main *challenges*: autonomous aerial interception of objects, cooperative robotic structure building, and autonomous robotic fire extinguishing. Many prestigious research institutes entered the competition, including ETH Zurich, KAIST, University of Bonn, University of Tokyo, University Carnegie Mellon, and CNRS, among others. Our group participated in all three of these challenges in cooperation with the University of Pennsylvania and New York University. My contribution was to Challenge 3, pertaining to fire extinguishing, which consisted of three sub-tasks in which my expertise was applied to all.

The most relevant contributions include the techniques I developed for aerial extinguishing of fires in two specific modalities - with aerial spraying of liquid extinguishant, and by aerial placement of fire blankets. I also participated tangentially on the sub-task of liquid-based fire extinguishing using a ground robot [39], but given the significantly better stability of a ground robot compared to an aerial robot, only the tasks involving UAVs became relevant to my further research on UAV-based VRL. In all of these sub-tasks, I have leveraged thermographic cameras and laser range sensors, allowing for accurate relative pose estimation of targets in terms of both their bearing and distance.

In the aerial liquid-based fire extinguishing tasks, there were two types of targets: "fire analogues", consisting of an aluminium heating element, and a *real gas fire ring* with a heater in the center. In both cases, the target objects had a liquid receptacle for scoring based on the amount of extinguishant deposited.

Although the fire analogues are comparatively easy to detect indoors using a thermographic camera with their distance retrievable using laser range sensors, extinguishing fires using liquid sprayed from a UAV still carried significant challenges pertaining to the vision-based control. Due to the combined effects of air currents and the pushback of the sprayed liquid, the firefighting UAV is subject to significant drifting. The relative drifting itself can be measured using the sensor system. If the UAV is currently not aiming from a suitable position at the target, it can either keep spraying in the incorrect direction, wasting the limited amount of stored liquid, or it can immediately stop spraying, allowing the fire to re-ignite, while also inducing more dynamic, difficult to compensate-for force feedback on the UAV. We therefore needed to ensure that the UAV was able to maintain continuous aim for extended

periods of time. The core of the unique stabilization problem here dwells in the fact that in order to correct the relative position of a partially-actuated[1] UAV with respect to a target, the UAV in question first needs to tilt. Doing so not only redirects the camera sensors used, potentially losing the line of sight, but it also redirects the spray nozzle. The solution I developed involved modifying the dynamic flight control to better suit the computer vision, as well as the spraying action. Rather than constantly controlling the spraying UAV to an "ideal" relative spraying position w.r.t. the target object, it was allowed a region of *hysteresis* where positional drifting was allowed, while only correcting the heading of the UAV. This made it possible to continuously spray the extinguishant at the target for extended periods of time, while also keeping the target in view of the cameras for more precise relative pose estimation. This technique is described in detail in the attached paper

V. Spurny, V. Pritzl, V. Walter, M. Petrlik, T. Baca, P. Stepan, D. Zaitlik, and M. Saska, "Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle," *IEEE Access*, vol. 9, pp. 15 872–15 890, Jan. 2021. DOI: `10.1109/ACCESS.2021.3052967`,

where the sub-task of extinguishing indoor fires is discussed. Attaching the nozzle and the camera of an active UAV to a gimbal can help with continous aim, but at the time of writing, these devices are still relatively costly, especially if they need to be able to withstand the considerable forces created by the flow of the extinguishant.

Extinguishing real fires had the above stabilization issue as well, while additionally complicating the targeting due to the visually-dynamic appearance of flames in both the thermographic camera images, as well as in the RGB camera images also used for this task. The flames moved and changed their brightness and temperature in response to air currents, especially those created by the presence of a multirotor UAV, as well as by changing pressures in the gas intake and the act of liquid-based extinguishing itself. The solution herein combined separate computer vision solutions for thermographic imaging and RGB imaging, exploiting the strengths of each, while compensating for the drawbacks inherent to each modality. Thermographic imaging makes it possible to locate very hot objects - especially active flames - from great distances. However, it also has drawbacks, including the lower resolution compared to equivalently priced RGB sensors and the fact that flames are opaque in this imaging method, obscuring other features useful for precise target localization. On the other hand, once the UAV was sufficiently close, RGB imaging allowed us to locate other target features for more precise aiming by exploiting the partial transparency of flames to specific visible wavelengths. This sub-task and the computer vision system is described in detail in the attached paper

V. Walter, V. Spurny, M. Petrlik, T. Báca, D. Zaitlík, L. Demkiv, and M. Saska, "Extinguishing real fires by fully autonomous multirotor UAVs in the MBZIRC 2020 competition," *Field Robotics*, vol. 2, pp. 406–436, Apr. 2022, ISSN: 2771-3989. DOI: `10.55417/fr.2022015.`

The task of aerial placement of fire blankets required a precise $\mathbb{R}^3 \times \mathbf{S}^1$ pose estimation of a target in order to achieve maximum possible overlap of the placed blanket on the target. This helped me to develop a set of tools that were later re-used in the pose estimation module of the *UVDAR* system. The interaction of the UAV motion with the camera view

---

[1]as opposed to specialized fully-actuated platforms capable of full 6DoF control

was addressed in this case by accumulating pose measurements for an extended period of time without moving before executing the rapid placement of a fire blanket based on prior measurements. This meant that the placement action following successful estimation occurred in the shortest time possible, in order to minimize the drift we would accumulate in the time without direct target observation. The rapid action also minimized potential contact of the UAV with fire. Tailoring the computer vision to the specific task at hand using meticulously collected environmental data was crucial, since the thermal appearance of the ground fire analogues turned out to be significantly different than our intuition held. Our solution to the blanket placement sub-task is described in the attached paper

V. Walter, V. Spurný, M. Petrlík, T. Báča, D. Žaitlík, and M. Saska, "Extinguishing of Ground Fires by Fully Autonomous UAVs Motivated by the MBZIRC 2020 Competition," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, © 2021 IEEE. Reprinted, with permission., Jun. 2021, pp. 787–793. DOI: `10.1109/ICUAS51884.2021.9476723`.

Overall, my work on the competition contributed to the excellent overall ranking our team achieved. By successfully competing in all three challenges, the CTU-UPENN-NYU team qualified to compete in the *Grand challenge* of the competition, winning first place[2].

In addition, the observations I made on estimation and temporal fusion of visually-observed targets proved invaluable in my subsequent work. The knowledge and tools gained were applied in both the relative pose estimation of neighboring UAVs and in addressing the problems inherent in using VRL for mutual relative localization of cooperating UAVs.

---

[2]https://web.archive.org/web/20210805164213/https://www.mbzirc.com/winning-teams/2020

IEEE *Access*

Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle

**VOJTECH SPURNY[ID], VACLAV PRITZL, VIKTOR WALTER[ID], MATEJ PETRLIK,
TOMAS BACA[ID], (Graduate Student Member, IEEE), PETR STEPAN,
DAVID ZAITLIK[ID], AND MARTIN SASKA, (Member, IEEE)**
Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, 160 00 Prague, Czech Republic

Corresponding author: Vojtech Spurny (vojtech.spurny@fel.cvut.cz)

ABSTRACT This paper presents a novel approach to autonomous extinguishing of indoor fires inside a building by a Micro-scale Unmanned Aerial Vehicle (MAV). In particular, controlling and estimating the MAV state, detection of a building entrance, multi-modal MAV localization during the outdoor-indoor transition, interior motion planning and exploration, fire detection and position estimation, and fire extinguishing are discussed. The performance of these elements, as well as of the entire integrated system, are evaluated in simulations and field tests in various demanding real-world conditions. The system presented here is part of a complex multi-MAV solution that won the Mohamed Bin Zayed International Robotics Challenge 2020 (MBZIRC 2020) competition, and is being used as the core of a fire-fighting Unmanned Aerial System (UAS) industrial platform under development. A video attachment to this paper is available at the website http://mrs.felk.cvut.cz/2020firechallenge-insidefires.

INDEX TERMS Unmanned aerial vehicle, autonomous systems, firefighting, mobile robots, rescue robots.

I. INTRODUCTION

Micro-scale Unmanned Aerial Vehicles (MAVs) are nowadays used in numerous applications due to their potential for rapid deployment and their ability to reach locations that are difficult or dangerous for humans to access [1]. Despite advances in the autonomy and the reliability of MAVs, they are most often still teleoperated by a pilot while helping on site after natural disasters. Teleoperated MAVs are used for various situations, e.g. for providing assistance for cities hit by an earthquake [2], [3], finding victims in urban areas [4], localizing flooded areas [5], finding survivors during floods [6], and quickly localizing forest fires [7]. Further examples of robots assisting in search and rescue missions are presented in [8].

The MAVs deployed in the applications mentioned above operate at high altitudes, where no obstacles can be encountered and Global Navigation Satellite System (GNSS) localization is reliable. However, to fully exploit the potential of MAVs assisting in disaster response tasks, it is necessary to

The associate editor coordinating the review of this manuscript and approving it for publication was Juan Liu[ID].
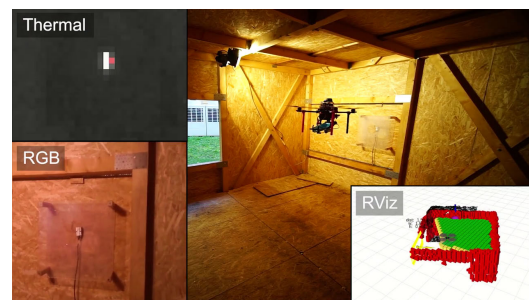
FIGURE 1. The proposed system uses processed data from the onboard sensors of the MAV towards the goal to extinguish fire inside the building.

move from teleoperated robots to autonomous robots that perceive their environment, can reliably localize and navigate in it, and furthermore, can influence their state by interacting with objects of interest. Autonomous MAVs have already been tested for use in locations where teleoperated MAVs cannot operate, e.g. coal mine tunnels [9], which can be dangerous to access after natural disasters such as earthquakes or

gas fires. A multi-modal mapping unit [10] can be attached to an MAV to provide autonomous exploration of GNSS-denied dark environments. A small agile MAV can be deployed as a first responder [11] of a rescue team to assess the situation in mines, to evaluate the risk of human rescuer injury and, most importantly, to find visual cues about the location of possible survivors in order to direct further rescue operations. During search and rescue operations, human rescuers underground often risk exposure to noxious gases. To reduce such hazards, a system described by [12] deploys an MAV with sensors capable of detection and measurement of the concentration of such gases. Multiple gas detecting MAVs deployed in parallel can be used for precise localization of the gas source [13]. Operations in places that are in the proximity of a source of radiation, such as the interior of a nuclear power plant, have to be planned with limited exposure time of human workers. The assistance of autonomous MAVs is therefore valuable in localizing the source of radiation [14], or in finding survivors [15], without risking prolonged exposure of human rescuers.

The approaches mentioned above were designed only for indoor environments or only for outdoor environments. However, the challenge and the novelty of the approach presented here is in the required transition from the open space around the building into the confined space of the rooms. These environments require different localization and state estimation techniques.

A cooperative firefighting mission called the Fire Challenge was the most complex task at MBZIRC 2020.[1] This challenge was motivated by the use of robots for urban firefighting, and it required a team of robots to collaborate on a series of urban firefighting-related tasks in outdoor and indoor environments. In this challenge, three MAVs and one Unmanned Ground Vehicle (UGV) had to collaborate to autonomously extinguish a series of fires (real and simulated) in an urban building. The fires were placed at various random locations at ground level in the arena (indoor and outdoor), and at different heights of the building. The challenge can be divided into four separate sub-tasks: extinguishing interior fires by a UGV, extinguishing fires on the facade of the building by MAVs, extinguishing ground fires by MAVs, and extinguishing fires inside the building by MAVs. These tasks were meant to be solved in their full scope, including searching for fires with unknown positions, fire extinguishing, and cooperation among multiple MAVs and a UGV working in the same environment. The deployment of such a team, as opposed to a single unit, was motivated by the requirement for minimal total mission time, as time is a critical factor for eliminating fire spreading and for saving people in real scenarios. Team deployment also enabled the use of a range of firefighting techniques (a fire-extinguishing agent or a fire blanket) and various platforms (MAV and UGV).

This paper addresses what we consider to be the most challenging task of the third challenge of the MBZIRC 2020

[1]http://www.mbzirc.com/

competition – fire extinguishing inside a building (see Fig. 1). The work presented here contributed not only to the MBZIRC 2020 challenge and to the firefighting mission but it also benefits other MAV applications. Research on indoor-outdoor transition through the narrow space of small windows and precise multi-sensor based servoing is important for a wide range of MAV challenges that are being tackled nowadays.

A. PROBLEM DEFINITION AND REQUIREMENTS ON THE PLATFORM

To solve the task of fire extinguishing inside a building, we assume that the size of the MAV platform, including the propellers, is limited by the width of the windows through which the platform has to fly. Of course, this width will be unknown during the deployment of the system in a real firefighting scenario. However, the organizers of the MBZIRC 2020 competition specified the window size to dimension of 2 m. Choosing the right platform size is crucial to task performance, as a smaller platform allows for a larger margin of error of the localization and control systems. On the other hand, a smaller MAV can carry less fire-extinguishing agent, and cooperation among multiple agents may be needed to extinguish a single fire.

We expect that the MAV will be equipped with a flight controller that commands the Electronic Speed Controllers (ESCs) to drive the brushless motors propelling the MAV, based on angular rate commands from an onboard computer. Furthermore, this flight controller should contain a set of sensors, such as accelerometers, gyroscopes, barometers, and magnetometers, and should provide them to the onboard computer for MAV state estimation. This onboard computer should provide sufficient computational power to solve all the required onboard processing tasks, in addition to besides general MAV control, state estimation, and collision-free motion planning.

For outdoor flying capability, the MAV has to be equipped with a GNSS receiver. However, the precision of the position data derived from satellite-based positioning systems can drift in the proximity of tall structures such as buildings, and this may block the visibility of some of the satellites, or may reflect the signal. The most fitting sensors that can be deployed to avoid a possible collision with the building are 3D LIDARs, thanks to their high information density and precise measurements of the obstacle distance. However, these devices are still relatively expensive and heavy. The proposed system therefore requires the MAV to be equipped with two complementary sensors — 2D LIDAR and a stereo camera. These sensors were selected because the data they provide can also be used for MAV control, for state estimation, and for collision-avoidance inside the building. Furthermore, this data is useful for correct detection of the window, for planning a collision-free trajectory through it, and also for estimating the position of detected fires.

The proposed system requires a thermal camera, or rather, a compound sensor consisting of multiple thermal cameras

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

for detecting fires. These cameras need to be arranged in such a way that the MAV flying at a safe distance from the wall will cover the whole wall, from the floor to the ceiling, with their Field Of Views (FOVs) to minimize the chance of missing a fire source. The MAV should be further equipped with two laser rangefinders. One faces downwards to measure the distance to the ground, and the second faces upwards to measure the distance to the ceiling when the MAV is inside the building. Data from these sensors is used for MAV altitude estimation and to help ensure safe flight. To extinguish the fires, the MAV has to be equipped with a water bag and with a pump that can force water through a nozzle mounted on the front of the MAV. To reduce the weight, the nozzle can be rigidly attached to the MAV, since it does not have to be actively stabilized.

We assume that the GNSS signal is available only outside the building, and not inside. The building interior contains unknown obstacles (*e.g.* a bed, a TV with a table, or a dining table), so that the system for fire extinguishing inside the building meets realistic assumptions. Lastly, we considered that a direct line of sight to the MAV would not be maintained during the whole mission, especially after entering the building, and that teleoperation through the base station would not be possible. Therefore, the task has to be solved completely autonomously, using only the onboard equipment of the MAV.

B. RELATED WORK AND CONTRIBUTION

Employing MAVs in firefighting has already been explored in several works. An obvious example of a situation where MAVs can prove beneficial is outdoor fire detection and monitoring. As reported in [16], a system of multiple MAVs can be used for automatic forest fire monitoring using visual and infrared cameras. Real experiments with forest fire monitoring in a national park have already been conducted by the Hungarian fire department [17]. The authors of [18] describe a task allocation strategy for distributed cooperation of ground and aerial robot teams in fire detection and extinguishing. In [19], an MAV system is designed to extinguish a fire by dropping a fire-extinguishing capsule on it.

MAVs fighting fires have also already been a topic for robotic competitions. [20] describes the design and implementation of a firefighting MAV for outdoor applications designed specifically for the IMAV 2015 competition. The employment of MAVs could also prove to be beneficial and life-saving in urban environments. Studies have already been done on fire detection in urban areas using a thermal camera carried by an MAV [21]. MAVs capable of entering buildings through doors and windows will be especially helpful, because of their ability to reach the target location much earlier than human firefighters. [22] contains the design of a semi-autonomous indoor firefighting MAV. The authors designed a fireproof, thermoelectrically-cooled MAV equipped with visual and thermal cameras, a collision avoidance module, and a first-person view system. However, to fully exploit the potential of MAVs in firefighting and to

achieve reliable operation, the MAVs themselves need to be autonomous.

MAVs autonomously entering a building through a window has already been partially explored in the literature, using various approaches with varying levels of experimental verification. In [23], the authors used RGB camera images and 2D LIDAR data for window detection and tracking, visual servoing while approaching the window, and potential field-based planning for the fly-through itself. However, their approach for window detection requires an operator to manually select a point of interest in the RGB image, and was verified only by simulations. This approach was further extended and verified in real-world experiments with flight through a wooden frame in [24]. However, the experimental verification only consisted of manually guiding the MAV in front of the window, autonomous flight through the window, and immediate manual landing. [25] utilizes stereo image pairs for detecting and estimating a window that can potentially be used by an MAV for entering a building. However, the proposed algorithm was verified only on data captured using a hand-held stereo rig. [26] deals with window detection from an RGB-D camera along with the generation of an optimal trajectory to a point in front of the window, but the approach was only verified in simulations. [27] focuses on cooperative control of an ornithopter MAV using visual servoing for narrow passage traversal. They demonstrated their approach in a real experiment with a small MAV flying through a narrow wooden frame. This approach requires a ground station continuously observing the scene throughout the flight. [28] deals with state estimation, control, and planning for an aggressive flight by an MAV through a narrow window tilted at various angles. State estimation is done based on visual camera images and an Inertial Measurement Unit (IMU). However, the position of the window needs to be known beforehand. Similarly, [29] focuses on an aggressive flight of an MAV through narrow gaps tilted at various angles employing a forward-facing fisheye camera for gap detection. A black-and-white rectangular pattern was used to simplify the detection of the gap. The approach for window detection and autonomous entering of buildings proposed in this paper was designed specifically for reliable performance under real-world conditions, and to function as a part of a complex autonomous system without input from a human operator. As such, the proposed approach was extensively verified in complex real-world experiments and therefore surpasses previous approaches both in the degree of autonomy under real-world conditions and in the complexity of its experimental verification.

Transition between an indoor environment and an outdoor environment creates the necessity to combine different localization methods in a single flight. [30] describes a system combining visual and laser odometry with IMU, using an Extended Kalman Filter (EKF) for flight in both indoor and outdoor environments. The authors use one common filter for fusing measurements from laser scan matching and from correlation-based visual odometry. The data source that is currently fused is determined by its variance. Fusion in one
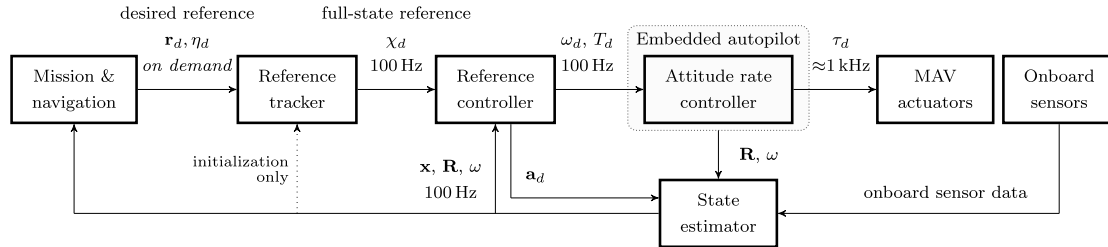
48

FIGURE 2. A diagram of the control system architecture. *Mission & navigation* software supplies a position and heading reference (r_d, η_d) to the reference tracker. The *Reference tracker* creates a smooth and feasible reference χ_d for the reference feedback controller. The feedback *Reference controller* produces the desired thrust and angular velocities (T_d, ω_d) for the embedded *Attitude rate controller*. The *State estimator* fuses control input a_d with data from the onboard sensors to create an estimate of the MAV translation and rotation (x, R, ω). For a more detailed description of the whole control pipeline, see [32].

common filter is problematic when the reference frames of the two odometry sources are not coincident, *e.g.* due to imprecise extrinsic calibration. Moreover, in the case of GNSS and LIDAR odometry, the frames of reference are inherently different. We propose fusing each type of measurement in its own separate filter and then choosing the better output to close the control feedback loop. [31] describes an approach that uses depth image processing for visual odometry capable of navigating MAVs during indoor and outdoor flight, and during transfers between these two types of flight. The solution relies on stereo camera depth estimation, which is much less precise than direct distance measurements using a laser sensor. Our solution offers a higher level of autonomy, as the whole mission is governed by a mission control state machine while in [31] the MAV is controlled by waypoints manually entered by an operator.

To sum up, the contributions of this paper are in the complexity and the reliability of the proposed system, which includes indoor outdoor transition, interaction with the environment based on vision from thermal cameras, precise MAV stabilization and control for safe flight through window, and for firefighting. Furthermore, the paper proposes a new approach for handling data from multiple sensors to robustly obtain a single state estimate – MAV pose estimation, height estimation, relative window pose estimation, and relative pose of the fire target estimation. All of these estimates are crucial for safe autonomous flight in complex MAV missions, and the proposed redundancy by using various sensors is necessary for achieving reliability required for industrial applications.

II. AUTONOMOUS SYSTEM DESIGN

The proposed system components are described in this section. Note that the entire system is run on the onboard PC only, allowing for full autonomy without any control station or teleoperation needed.

A. CONTROL AND ESTIMATION OF THE MAV STATE IN OUTDOOR AND INDOOR ENVIRONMENTS

One of the main contributions of this paper is a system that allows precise control for flying through relatively small windows and for inserting water into a small opening of a measurement device using multi-sensor control feedback. Additionally, in a real scenario, precise placement of the fire fire-extinguishing agent is crucial for mission success. Another important aspect of the system is the MAV state estimation approach, which allows precise localization and stabilization in the open space around the building, inside rooms with obstacles, and a smooth transition between these work-spaces.

The MAV is controlled by the novel multi-layer control pipeline, depicted in Figure 2, which was suited for the proposed system using the general control framework presented in [32]. The desired trajectory reference is supplied by higher-level motion planning modules that are specialized for each mission phase, as described in II-F. The reference is first processed by the *Reference tracker* [33], based on model predictive control to obtain a smooth and feasible reference for the *Reference controller*. The tracker also imposes constraints on the MAV states to prevent fast and aggressive motions, which are undesirable when navigating constrained indoor environments. The *Reference controller* uses the processed reference to provide SE(3) geometric state feedback control [34] of the translational dynamics and the orientation of the MAV. This type of controller achieves minimal control errors, which allows precise window flythrough. The attitude rate and thrust commands generated by the *Reference controller* are sent to the embedded *Attitude rate controller* in the flight control unit of the MAV, which controls the speed of each motor, using ESCs. The feedback loop of the *Reference controller* is closed by the *State estimator*, which fuses data from onboard sensors with the MAV altitude to obtain a precise and reliable state estimate for both the indoor and the outdoor phase.

The state estimation process uses Kalman filtering to estimate the 3D position of the MAV and its heading angle, along with their respective first and second derivatives. The MAV state is divided into lateral, altitude, and heading parts. Such decoupling facilitates tuning of the filters. Smaller system matrices save computation resources, allowing for running multiple filters in parallel. All active filters are grouped in a filter bank, visualized in Figure 3, from which the best filter

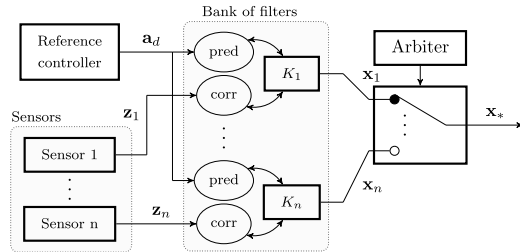# CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

FIGURE 3. The bank of filters $K = \{K_1, \ldots, K_n\}$. The prediction step is driven by the desired acceleration a_d. The correction step is triggered asynchronously as sensor measurements z_i, where $i = \{1, \ldots, n\}$, arrives. The output hypothesis x_* is chosen by the arbiter.

(a) View from an exter- (b) Depth image from (c) Visualization of the
nal camera. the onboard camera. data.

FIGURE 4. Example of window detection using the stereo camera and 2D LIDAR. The red squares in the visualization represent the LIDAR data, the blue rectangle represents the depth detection, and the yellow rectangle represents the filtered estimate.

for the current situation is used to close the feedback loop of *Reference controller*.

Since the fire-extinguishing mission consists of two phases (outdoor and indoor), the bank of filters for the lateral axes contains one for each phase. Both filters have the same three-state model with the desired acceleration from *Reference controller* on the input. The difference between the filters is in the sensor measurement used to correct the state in the update step of the Kalman filter iteration. The outdoor filter uses position corrections from GNSS and heading corrections from magnetometers. Inside the building, GNSS cannot be used. Both position corrections and heading corrections are therefore provided by the Hector SLAM [35] algorithm, using 2D LIDAR. In general, the bank of filters may fuse multiple sensors and localization techniques for state estimation (e.g. 3D LIDAR, VIO, Optic Flow). One of the simultaneously running filters is always selected as the main estimator, depending on the reliability of the filter state estimation (estimation covariance) and based on consideration of the current environment. For the task solved in this paper, the high-level planner changes the main lateral estimator during transitions between indoor and outdoor phases, just before flying through a window. This approach prevents measurements from the sensor that are inappropriate for the current surroundings to corrupt the state estimate used for feedback control. Greater reliability is thus ensured during the window flythrough, thanks to lower position drift. The estimation framework is also responsible for synchronously broadcasting the change of main estimator, so that the tracker and the controller can react by updating their internal state accordingly. The switch of the main estimator is smooth and seamless without producing any spikes in the controller output, which makes it virtually unnoticeable.

The altitude estimation fuses data from a barometer with measurements from a laser rangefinder. The rangefinder measures the height above the terrain, which can result in sharp changes in height when flying above objects protruding from the ground plane. Measurements of this type are declined by a median filter when the tall object is visible for less than 1 s and, in combination with the barometer data fusion,
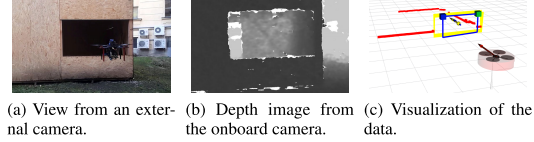
a smooth altitude estimate suitable for control is obtained. In the case of a height measurement jump lasting for more than 1 second, the measurement is offset by the difference from the original height. Up until this time, the median filter suppresses reaction to the measurement change.

B. WINDOW DETECTION AND ESTIMATION

In order to enter the target building, a suitable entrance needs to be detected and its position needs to be continuously estimated during the entire flythrough. For this purpose, depth data from the stereo camera and 2D LIDAR data are utilized. The depth data can provide complete information about the 3D position, the orientation, and the size of the window. However, these detections contain significant noise and the FOV of the stereo camera is limited. On the other hand, the 2D LIDAR data is very precise and 2D LIDAR is capable of seeing the window during the whole flythrough maneuver, regardless of the MAV orientation. The main drawback of 2D LIDAR is that it only provides information about the window in the 2D horizontal plane. The proposed approach offers multiple modes of estimation. In *depth + lidar* mode, the depth data is fused with the 2D LIDAR data with no apriori-known information required. In *lidar + apriori* mode, only 2D LIDAR detections and apriori-known information about window size and altitude are used for window detection. This mode was used to simplify window detection for the competition, in which information about window size and altitude was made available beforehand. Both modes work only in the case that the window is not obstructed.

Window detection from the depth data is described in Algorithm 1. The algorithm takes the 1280×720 depth image published by the stereo camera and downsamples it by a factor of 8 to reduce the computational demands on the CPU. The algorithm then searches for contours in the image and tries to fit quadrilateral shapes with certain parameters to the data. After identifying such quadrilaterals, a check is performed to ensure that the detected shape is an opening and not a protrusion. The algorithm starts from the center of the quadrilateral, where it generates a Region of Interest (ROI) such that its aspect ratio is the same as the initial quadrilateral and the shorter size is two pixels. It then checks if the depth of all the pixels within this ROI is greater than the plane of the initial quadrilateral. Then it iteratively expands this ROI and repeats the check until the check fails. Afterwards, the corners of the last ROI are projected to the plane of

---

Algorithm 1 Detection of Window From Depth Data

Input: Raw depth image $\mathcal{I}_{raw}$
Output: List $\mathcal{D}_{depth}$ of detected windows
1: **function** Detect_windows_depth($\mathcal{I}_{raw}$)
2:      $\mathcal{I}_{down} \leftarrow$ Downsample($\mathcal{I}_{raw}$)
3:      $\mathcal{C} \leftarrow$ Find_contours($\mathcal{I}_{down}$)
4:      **for** $c \in \mathcal{C}$ **do**
5:          $q \leftarrow$ Fit_quadrilateral(c)
6:          **if** $q \neq \varnothing$ and Is_hole(q) **then**
7:              Add q to $\mathcal{D}_{depth}$
8:      **return** $\mathcal{D}_{depth}$ ▷ Detections are passed to the LKF

---

the initial quadrilateral. Then the size of the quadrilateral formed by the projected points is compared with the size of the initial quadrilateral. If it is above a certain ratio, the new quadrilateral is accepted as a traversable window. Otherwise, the detection is discarded.

Window detection from 2D LIDAR data is described in Algorithm 2. Firstly, a combination of line extraction algorithms is utilized to identify possible window edges. The Successive Edge Following algorithm [36] is used for detecting window edge candidates $\mathcal{E}_1$ based on sudden changes in 2D LIDAR measurements. The algorithm parses the original scan into a set of segments $\mathcal{S}_1$ by splitting the scan in places where the difference between two consecutive measurements exceeds a predefined threshold. The endpoints of the segments $\mathcal{S}_1$ are used as window edge candidates $\mathcal{E}_1$. The set of segments $\mathcal{S}_1$ is then passed to the Iterative End-Point Fit algorithm [36]. The algorithm fits a line through the endpoints of each segment and splits the segment into two sub-segments at the point most distant from the line, if the distance exceeds a predefined threshold. This process enables us to detect the window edges in the form of a corner protruding towards the MAV, which occurs when a wall or an obstacle is located right next to the window inside the building. Depending on the mode of estimation, the final detections are then produced either by linking the window edge candidates $\mathcal{E}$ to existing window estimates $\mathcal{W}$ initialized based on depth detections, or by standalone detection based on apriori-known information $\mathcal{A}$ describing window size and altitude.

The Linear Kalman Filter [37] is used for fusion of the individual detections and for filtering out measurement noise. The state **x** of the Kalman filter describing a single window is defined as

$$\mathbf{x} = \begin{bmatrix} c_x, c_y, c_z, \phi, w, h \end{bmatrix}^T , \qquad (1)$$

where c_x, c_y, c_z are Cartesian coordinates of the window center, $\phi \in [-\pi, \pi]$ is the angle between the projection of the normal vector of the window to the xy-plane and the x-axis (i.e., rotation around the z-axis), w is the width, and h is the height of the window. It is assumed that the window is not tilted and is perpendicular to the ground plane. The position and the orientation of the window are specified in

---

Algorithm 2 Detection of Window From 2D LIDAR

Input: List $\mathcal{P} = \langle \boldsymbol{p}_1, \ldots \boldsymbol{p}_n \rangle$, where $\boldsymbol{p}_i$ are points obtained from a single laser scan; *mode* $\in \{depth + lidar, lidar + apriori\}$ - selected mode of estimation; (optional) list $\mathcal{W}$ of existing window estimates; (optional) list $\mathcal{A}$ of apriori information
Output: List $\mathcal{D}_{lidar}$ of window edge pairs
1: **function** Detect_windows($\mathcal{P}$)
2:      $\mathcal{E}_1, \mathcal{S}_1 \leftarrow$ SEF($\mathcal{P}$)   ▷ Successive Edge Following
3:      $\mathcal{E}_2, \mathcal{S}_2 \leftarrow$ IEPF($\mathcal{S}_1, \mathcal{P}$)   ▷ Iterative End-Point Fit
4:      $\mathcal{E} \leftarrow \mathcal{E}_1, \mathcal{E}_2$
5:      **if** *mode* $= depth + lidar$ **then**
6:          $\mathcal{D}_{lidar} \leftarrow$ Link_edges_to_estimates($\mathcal{E}, \mathcal{W}$)
7:      **else if** *mode* $= lidar + apriori$ **then**
8:          $\mathcal{D}_{lidar} \leftarrow$ Standalone_detection($\mathcal{E}, \mathcal{A}$)
9:      **return** $\mathcal{D}_{lidar}$ ▷ Detections are passed to the LKF

---

a global coordinate frame and therefore all the state vector elements are modeled as static. Figure 4 shows an example of window detection for a mock-up building constructed at the Czech Technical University in Prague. The figure contains an external view of the MAV hovering in front of the window, a depth image from the onboard camera, and a visualization of the data.

C. INDOOR MOTION PLANNING AND EXPLORATION

The position of the fire is unknown (in the competition and usually also in a real application) before the mission and therefore the interior must be explored to find its location. Obstacles can be detected using data from the stereo camera and from 2D LIDAR. We store the information about the environment around the MAV in an occupancy grid within a three-dimensional buffer that slides along with the MAV. Our buffer is a modification of the implementation of a buffer that has been developed as a part of the *Ewok* system [38]. Originally, the buffer was in the shape of a cube, with the same number of blocks in all axes. However, the height of a room is usually much smaller than its horizontal dimensions. Therefore, we modified the buffer to be able to specify its z-axis dimension separately. For the experiments, the buffer size was set to $128 \times 128 \times 32$ voxels with resolution of 0.1 m. This improves the times for inserting the data from the sensors and for recomputing the buffer in the case of MAV movement. See an example of a visualization of the occupancy grid in Figure 5(a).

The trajectory planning in our case is done using an A* planner that finds a local plan in a three-dimensional grid with a predefined minimal allowed distance from the obstacles (the grid has the same resolution as the occupancy grid). This plan is further processed by removing redundant points, in order to smooth out the overall path. The points that can be removed are those where the path between the previous point and the next point in the plan is still safe. This is done iteratively until no points can be removed from the plan. Because the

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES
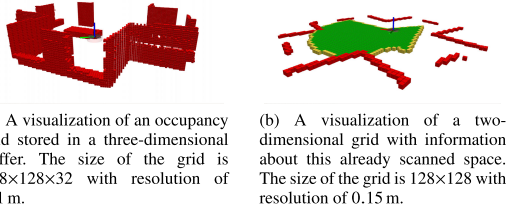
(a) A visualization of an occupancy grid stored in a three-dimensional buffer. The size of the grid is 128×128×32 with resolution of 0.1 m.

(b) A visualization of a two-dimensional grid with information about this already scanned space. The size of the grid is 128×128 with resolution of 0.15 m.

FIGURE 5. Examples of occupancy grids around the MAV used for motion planning and exploration inside the building. The red elements denote obstacles, the green elements show the already scanned space, and the yellow elements denote frontiers for exploration.

FIGURE 6. Thermal vision outputs. From left to right: Thermal camera view scaled from 0 to 120 °C, differential image produced with the Laplacian operator (scaled between the two extreme values of the image), and thresholded detection with the rounded centroid in red.

stereo camera at the front has a limited FOV, movement in any direction is allowed only when the MAV is facing in that direction. The plan is then sampled according to the permissible dynamics of the MAV for planning procedure and send to the reference tracker. The dynamic of the MAV for the planner was set for all deployments presented in Section III to a speed of $0.3 \, \text{m s}^{-1}$ and a heading rate of $0.3 \, \text{rad s}^{-1}$. This is within the capabilities of the MAV platform that had flying dynamic constraints set to a speed of $2 \, \text{m s}^{-1}$ and a acceleration of $10 \, \text{m s}^{-2}$. This ensures that the processed trajectory produced by the reference tracker will not differ from the plan by an unsafe amount.

One of the primary goals for successfully completing the mission is to find the fires. Using the planning presented here, we can fly without collision inside the building. However, it is necessary to specify the position that we want to reach. To ensure that each room is completely scanned for fire sources, a novel exploration algorithm had to be designed. The whole exploration process is described in Algorithm 3. The proposed approach differs from state-of-the-art exploration methods, which are designed to build a map in which the robot is localized. In our case, one set of sensors is applied for simultaneous localization and mapping, and a different set is applied for fire source localization. Moreover, the FOV of the two sensory sets differs significantly, requiring different exploration strategies. The exploration strategy relies on information about which parts of the interior have already been seen by the thermal cameras. The part that has already been visited (i.e. visually scanned) is the space that was within the FOV of the thermal cameras and is closer than the maximum detection distance. We assume that the height of the room is completely covered by the vertical FOV of the thermal cameras. The exploration can therefore be simplified and solved as a two-dimensional problem, where the z-coordinate of the goal position is set to a constant flying height (the z-coordinate of the center of the opened window). For this purpose, a two-dimensional occupancy grid is built from the incoming sensor data. The horizontal FOV of the thermal cameras is projected to this grid, where all elements of the grid located within this field and unobscured by obstacles are updated. When an element has been updated thirty times (the replanning was set to 10 Hz, meaning that
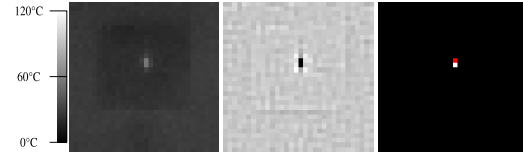
the block has been observed for 3 s), the block is marked as scanned. Frontiers - elements that are marked as scanned and that have at least one neighboring unscanned element - are then candidates for goal positions for the planning. From these frontier elements, we select the element that can be scanned the earliest and which has at least twenty unscanned elements in its proximity. If the selected goal position can be seen from the current position of the MAV merely by changing the MAV heading towards it, that plan is then used. Otherwise, a plan found by the trajectory planning is selected. An example of a visualization of a two-dimensional grid with information about the already scanned space and its frontiers is shown in Figure 5(b).

D. DETECTION AND ESTIMATION OF THE POSITION OF FIRES

It would be too complicated to safely create real fires inside a building. Therefore, the organizers of the competition decided to use artificial fires, hereafter referred to as fire analogues. These fire analogues are distinguishable both thermally and visually. The main body is made from plexiglass and is divided into two separate parts. The first part contains an anodized aluminium heating element with dimensions of 60 mm × 35 mm heated to 120 °C. This part is accessible through a 150 mm wide circular opening, and the task of the MAV is to spray water through this opening. The second part contains a silk flame that visually emulates flames and allows spectators of the competition to see which of the fire analogues was active. The second part is placed behind the first on the side opposite to the opening. Examples of what the fire analogues looked like during the competition are shown in Figure 11.

For effective extinguishing, it is necessary not only to detect the heating element in a thermal image, but also to detect its relative position w.r.t. the MAV in 3D. It is also necessary to obtain an estimate of the normal vector of the front plate of the fire analogue object in order to select the optimal extinguishing position for the MAV.

The heating elements are heated to 120 °C, but our thermal cameras[2] report them as being at a temperature of only approximately 70 °C. This is due to the material having an emissivity value of 0.55 [39], as opposed to the value

---

[2]https://terabee.b-cdn.net/wp-content/uploads/2020/05/evo-thermal_specsheet.pdf

---

Algorithm 3 Indoor Exploration

Input: horizontal FOV of the thermal camera ϵ

1: **function** Explore_interior(ϵ)
2:     $\mathcal{B}_{plan} \leftarrow [\ ]$ ▷ initialize 3D grid for planning
3:     $\mathcal{B}_{expl} \leftarrow [\ ]$ ▷ initialize 2D grid for exploration
4:     **while** not Fire_detected() **do**
5:         $\mathcal{R} \leftarrow$ Get_current_state_of_MAV()   ▷ $\mathcal{R} = (\mathbf{x}, \mathbf{R})$
6:         **if** Received_new_data_from_sensors() **then**   ▷ From the stereo camera or 2D LIDAR
7:             $d_{new} \leftarrow$ Get_new_data()
8:             $\mathcal{B}_{plan} \leftarrow$ Update_planning_buffer($\mathcal{B}_{plan}, d_{new}, \mathcal{R}$)
9:             $\mathcal{B}_{expl} \leftarrow$ Update_exploration_buffer($\mathcal{B}_{expl}, d_{new}, \mathcal{R}, \epsilon$)
10:        **if** Is_time_to_replan() **then**
11:            $f \leftarrow$ Get_nearest_frontier($\mathcal{B}_{expl}, \mathcal{R}$)
12:            **if** Is_empty(f) **then**
13:                **return** False   ▷ Space has been explored without detection
14:            $Plan \leftarrow$ Plan_trajectory($\mathcal{B}_{plan}, \mathcal{R}, f, \epsilon$)   ▷ $Plan = [\mathbf{r}_1, \eta_1, \ldots, \mathbf{r}_n, \eta_n]$
15:            Fly_trajectory($Plan$)
16:     Hover()   ▷ Stop following the previous trajectory
17:     **return** True   ▷ Fire detected

---

of 0.95 that our cameras internally use in calculating all surface temperatures. This means that the contrast between the heating elements and their surroundings in the thermal image is less than would be expected based on the temperatures alone (see Figure 6 on the left). This, however, was not a significant issue indoors, where the environment does not contain objects of such a temperature, and the viewing distances are short due to the limited size of the interior.

We could therefore merely detect the fire sources in the thermal image by binarizing the image with a fixed threshold lower than the typically measured temperature of the heating elements (Figure 6, right). In order to avoid detecting a uniformly heated background, such as a sun-heated wall, we additionally validated the detections using a differential image produced by the Laplacian image operator to check if the detected object is significantly hotter than its surroundings (Figure 6, center). It should be noted that the plexiglass casing of the fire analogues is not transparent to the infrared radiation used by our thermal cameras. This means that the observation angles, w.r.t. the front wall from which they can be seen, are limited. The observation has to be made and the extinguishing has to be done from a position as close to perpendicular with the wall. Since the radius of the circular front opening in the plexiglass casing is 7.5 cm and the heating element is positioned *approx.* 6 cm inwards from the front plate, the heating element can be seen from at most 57° from the perpendicular position. However, this is an extreme where we would only have a line of sight to the very edge of the heating element, which may not even appear on the camera. It is still desirable to maintain perpendicular alignment for extinguishing, since this maximizes the image area of the heating element, minimizes the influence of the parallax between the heating element and the front plate to

which we measure the distance, and additionally maximizes the robustness of the correct aiming w.r.t. drifting in an arbitrary direction.

We assume that the thermal cameras have the properties of pinhole cameras and derive their focal distance from the pixel resolution w per side and their FOV ϵ per side. This assumption is based on the minuscule size of the cameras, the small FOV, the relative rarity, and the high cost of infrared compatible lenses. More precise calibration than this is impractically complicated to achieve, due to the low resolution of the cameras.

When a contour of compliant pixels is detected in the thermal image, we calculate the average *x-y* image coordinates of these pixels. These coordinates are converted to direction vectors using the assumed camera model:

$$\mathbf{v_t} = \begin{bmatrix} v_{t_x} \\ v_{t_y} \\ v_{t_z} \end{bmatrix} = \begin{bmatrix} 1/f & 0 & -((w-1)/2)/f \\ 0 & 1/f & -((w-1)/2)/f \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad (2)$$

where

$$f = \frac{(w/2)}{\tan(\epsilon/2)}. \quad (3)$$

For convenience in subsequent operations, vector $\mathbf{v_t}$ is normalized and is transformed into a coordinate frame centered in the optical center of the camera, with the *x*-axis pointing forwards, the *y*-axis to the left, and the *z*-axis upwards. This coordinate frame is called the thermal base frame. The transformed vector is denoted as $\hat{\mathbf{v_f}}$.

For extinguishing action carried out by MAVs, we also need a distance estimate. This is achieved by combining the direction vectors with a surface shape measurement source. We use the 2D LIDAR sensor to estimate the outline of the fire analogues in front of the camera in the form of a set of

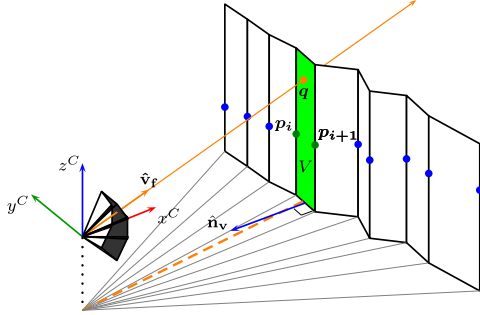# CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

V. Spurny *et al.*: Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle

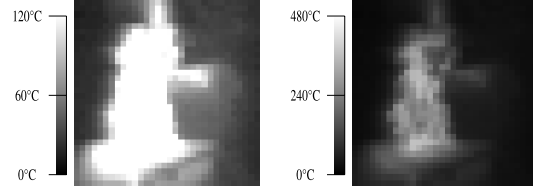**FIGURE 7.** Illustration of how the fire is localized.

FIGURE 8. An example of the view of a real fire with the thermal camera used in the competition at two different thermal ranges. Note the high contrast of the fire compared to the background, in addition to the large size of the fires in the image. In the range of temperatures used in the competition, the fire itself is completely saturated in the image. Detecting and targeting such objects is significantly easier to achieve than when fire analogues are used.

coplanar 3D points $\mathcal{P}_{lidar}$ ordered by their angle from the sensors (blue and green points in Figure 7). This works by presuming that the entire scene is a vertical extrusion object with equal cross-sections at all heights. We can then represent the scene as a set of vertical planes passing through pairs of 3D points adjacent in the sense of angle (white and green vertical planes in Figure 7). We select a pair of points p_i and p_{i+1} from $\mathcal{P}_{lidar}$, s.t.

$$\text{atan2}(p_{i_y}, p_{i_x}) > \text{atan2}(v_{f_y}, v_{f_x}), \tag{4}$$

$$\text{atan2}(p_{i+1_y}, p_{i+1_x}) < \text{atan2}(v_{f_y}, v_{f_x}). \tag{5}$$

The 3D position of the estimated target is then obtained by calculating the intersection of the optical line (the orange line in Figure 7) with the vertical plane V (the green plane in Figure 7) passing through the selected pair of points. Plane V is defined by point p_i and normal $\mathbf{n_v}$:

$$\mathbf{n_v} = \begin{bmatrix} -(p_{i+1_y} - p_{i_y}) \\ p_{i+1_x} - p_{i_x} \\ 0 \end{bmatrix}. \tag{6}$$

The intersection point q is then calculated as:

$$q = \hat{\mathbf{v}}_\mathbf{f} \cdot t, \tag{7}$$

where t is obtained using the normalized vector of the surface normal $\hat{\mathbf{n}}_\mathbf{v}$ as:

$$t = \frac{\hat{\mathbf{n}}_\mathbf{v} \cdot p_i}{\hat{\mathbf{n}}_\mathbf{v} \cdot \hat{\mathbf{v}}_\mathbf{f}}. \tag{8}$$

The estimate of the surface normal $\hat{\mathbf{n}}_\mathbf{v}$ and the intersection point q are used to steer the MAV into perpendicular position for extinguishing s, defined as

$$s = q + r_d \cdot \hat{\mathbf{n}}_\mathbf{v}, \tag{9}$$

where r_d is the desired extinguishing distance of 1.5 m.

We did not consider a single estimate of the 3D position of the heat source to be sufficient. Instead, we implemented a Kalman filter that stores multiple measurements as an array of states and refines each state using new measurements. The state vector of the Kalman filter used here is

$$\mathbf{x}_i = \begin{bmatrix} c_{i_x}, c_{i_y}, c_{i_z}, \psi_i \end{bmatrix}^T, \tag{10}$$

where c_{i_x}, c_{i_y}, and c_{i_z} are the coordinates of the fire in the world coordinate frame. The ψ_i represents the azimuth of the surface normal for that fire. The filtering mechanism stores multiple such state vectors, corresponding to multiple different detected fires. We update a specific state vector $\mathbf{x}_i$ using a new estimate of the fire position q and normal $\mathbf{n_v}$, if q is closer in the world frame than 1 m to $\begin{bmatrix} c_{i_x}, c_{i_y}, c_{i_z} \end{bmatrix}^T$ and at the same time the horizontal component of $\mathbf{n_v}$ is closer than 90° to ψ_i. If no such state is found, a new state is initialized based on the current estimate. As is typical of the Kalman filter, the state covariances grow in time to reflect loss of knowledge without observation and, in our case, additionally discard any measurements or even states that exhibit erroneous properties or states that have not been updated with a measurement for the past 10 s. To account for random errors, a state $\mathbf{x}_i$ is only used in fire extinguishing if it has been associated with at least 10 measurements.

It should be noted that the method for thermal detection and localization of fire analogues used here is significantly more complicated than would be realistically required in extinguishing real-world fires. This is because fires that pose a real danger are significantly hotter than the heating elements of the fire analogues used in the competition, and dangerous fires do not appear colder than they actually are in thermal cameras (see Figure 8 for an example). Additionally, these fires are significantly larger objects, and extinguishing them would not require just spraying into a very narrow opening. In real fire extinguishing, it is desirable to aim at the hottest areas detected in the fire. A more dispersed water stream would also be advantageous, as it would be more likely to hit the fire, even with less precise target localization, and it would extinguish the fire more effectively.

E. FIRE EXTINGUISHING
Upon obtaining the first validated fire detection state in the Kalman filter array, the MAV is sent to a position s 1.5 m in front of the given target q along the estimated normal $\mathbf{n_v}$. As the MAV flies there, its estimate of the target position and the surface normal improves as it obtains new detections from better viewpoints. When it reaches the position, control is handed over to the fire-extinguishing subsystem (the state labeled *Extinguish fire* in Figure 10(d)).
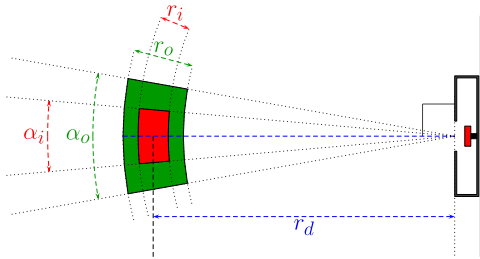
FIGURE 9. Steering hysteresis for fire extinguishing, viewed from above.

In this state, the MAV is steered towards position s. However, continuous positional stabilization of the MAV towards this exact position may lead to rapid tilting, due to movement oscillations during attempts to correct the current position, especially in face of potential fire estimation errors and disturbances such as air currents. This causes the direction of the water stream to be significantly unstable, since the water nozzle does not have active stabilization due to weight restrictions.

To mitigate this effect, we have included a hysteresis to horizontal steering through two ranges of angle and distance offsets (see Figure 9). The angle ranges were defined as limits to the angles formed by the surface normal $\mathbf{n_v}$ and the line connecting the center of the fire analogue with the MAV - the inner range α_i was set to $\pm 5°$, and the outer range α_o was set to $\pm 10°$. The distance ranges are offsets from the desired extinguishing distance of 1.5 m. These were set to ± 0.075 m for the inner range r_i, and to ± 0.15 m for the outer range r_o. Once the MAV has reached the inner ranges (the red zone in Figure 9), the xy-coordinates are not corrected, irrespective of disturbances. The system can only correct its altitude and its heading as changing these does not generate tilt of the MAV. The MAV thus tends to drift or "float". It is necessary to correct the heading continuously, otherwise the drifting would affect the aim of the MAV. Water spraying is only activated when the MAV is in this drifting state. The z-coordinate and the heading are controlled either to spray at the directly observed heating element of the fire analogue, or - if it is not visible *e.g.* due to being cooled down by the water - to spray at its estimated position from the Kalman filter. If the target is directly observed, the aiming is more responsive to disturbances. However, if the aiming relies on the filter, the precision is lowered. The MAV is only allowed to correct its xy-coordinates again when it has been moved outside the outer ranges (the green zone in Figure 9), at which point water spraying is disabled.

F. HIGH-LEVEL BEHAVIOR CONTROL

The complete behavior structure of the proposed system is constructed as a hierarchical state machine, which is used for interconnecting all the subsystems. The state machine was designed for robustness of the entire code structure by

resolving the remaining few subsystem failure cases due to wrong sequential and concurrent operations. The hierarchical state machine is implemented using the Flexbe library [40], and it is fully integrated into the designed ROS framework.

In Figure 10, the internal states of the state machines are visualized as single-outline rectangles, and the nested lower-level state machines are visualized as double-outline rectangles. Transitions between two states and from one state machine to a lower-level state machine are marked by arrows with labels of outcomes describing the given transition. Dotted terminal states represent the transition that is called after returning to a higher level state machine. A landing event is called whenever any state produces an outcome that means that the MAV cannot continue its mission. Unfortunately, there is no information available for the MAV to recognize whether the amount of the sprayed liquid was sufficient to extinguish the fire. Therefore, whenever the MAV lands, the operators can see whether or not the mission was successful by the state of the water bag.

The diagram of the main state machine is visualized in Figure 10(a). In the first step, the correct performance of all key parts of the system is checked. When every component is verified to be operational, an automatic takeoff is called. Once the MAV is in the air, the mission commences. The mission is divided into two parts: the outdoor phase and the indoor phase. The outdoor phase is the part of the mission where the MAV is outside the building. The indoor phase is when the MAV is inside the building. At the end of the mission (a window or a fire has not been found, or a fire has been successfully extinguished), the MAV flies back to the home position and lands.

The outdoor phase (Figure 10(c)) starts by flying to the known GNSS position of the building. This position must be a position from which the MAV is capable of detecting the building. A common problem with navigation using standard GNSS is its precision, which depends on the quality of the satellite signal. GNSS satellites broadcast their signals in space, but what we receive depends on additional factors, such as signal blockage and atmospheric conditions. Therefore, a safe position in front of the building may drift into the building. For this reason, the MAV uses scans provided by 2D LIDAR during the flight to facilitate navigation around the building where the GNSS quality is degraded by the building blocking the direct line of sight of some satellites. These scans provide planar information in 360° around the MAV and are used in a virtual bumper. The virtual bumper is a system that prevents the MAV from following a plan that would lead it to go closer than the predefined safe distance from the building. If the target position is inside the building, the MAV will stop at a position within a safe distance from the building and closest to the target position.

After it reaches a safe position near the building, the MAV starts flying alongside the building at a predefined distance with a heading towards the building, and begins the window detection mechanism. Whenever a window is located, the MAV stops flying alongside the building and flies in front

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES
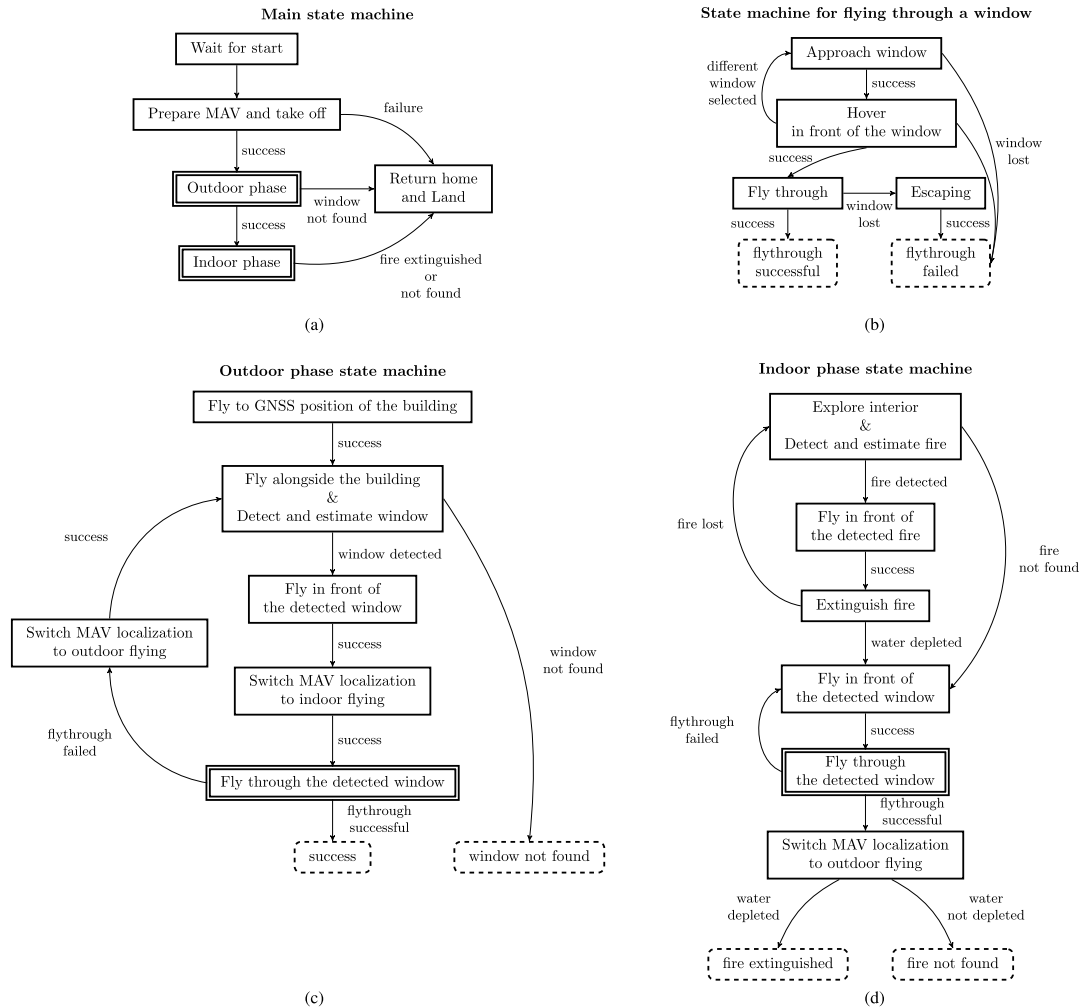
FIGURE 10. Diagrams of the main parts of the proposed system.

of the window to distance of 2 m from its center. Once this position is reached, the localization of the MAV is switched to indoor flying mode (LIDAR-based odometry is used in the controller feedback), and an attempt is made to fly through the window. If the attempt is successful, the MAV is inside the building and the outdoor phase is considered successfully finished. In the case of the opposite result, the MAV restarts the search for an open window. After circumnavigating the building without detecting a window, the outdoor phase ends, and the result is registered as "window not found". The MAV is allowed to detect the same window again and to attempt to fly through it. This strategy is motivated by the knowledge that only a single window was to be opened on

each floor during the competition trial. The attempts can be repeated until the maximum allowed flight time is reached. After reaching this time, the MAV automatically lands.

The steps for flying through the selected window are shown in Figure 10(b). The procedure utilizes window estimates produced by algorithms described in section II-B. First, the MAV flies to a position in front of the window while continuously facing the center of the window. The MAV then hovers in front of the center of the window to stabilize itself before the actual flythrough. The flythrough maneuver is then initialized and the state machine waits for an up-to-date window estimate corrected by new detections. After the window estimate has been updated, the MAV flies through the center of the

FIGURE 11. Photos of the MBZIRC 2020 Fire Challenge area. The photo on the left shows the tall structure simulating a building, the photo in the middle shows front view of the fire analogues (this unit was turned on) and the photo on the right shows side view of the fire analogues (this unit was turned off).

FIGURE 12. Snapshots from the simulation developed for the Fire Challenge of the MBZIRC 2020 competition.

window to a goal position at a predefined distance behind the window while maintaining a constant altitude. If the window estimate is lost while the flythrough is in progress and the MAV is still outside the building, the state machine switches to the Escaping state and the MAV returns to its original hovering position in front of the window.

The indoor phase (Figure 10(d)) contains the final parts – localization and extinguishing of the fire. Localization is done by using the exploration method, which is described in section II-C, and the detection system, which is described in section II-D. Once the fire is detected, the MAV flies in front of it and begins extinguishing (section II-E). If the fire target is not lost, the MAV depletes all the water that it is carrying during the extinguishing maneuver. There is no feedback that provides information as to whether the extinguishing has been completed. The extinguishing is therefore declared completed once all the water is depleted. In the case that the fire is lost, the MAV starts exploring again. After depleting the extinguishing agent, the MAV flies back in front of the window that it entered through and tries to fly back outside the building. An attempt to leave the building using the same window is also performed if the exploration finishes without successful localization of the fire. When the MAV is outside, the localization of the MAV is switched to outdoor flying again and MAV flies back to land on the starting position.

In the case of a real firefighting scenario inside a building, the proposed system can be used in almost the same structure as presented here. The only modification is that the process of searching for an open window can be accelerated by directly specifying the approximate GNSS position of the window.

III. EXPERIMENTAL RESULTS
A. SIMULATIONS
To be able to experimentally verify the entire firefighting mission, we modeled the MBZIRC 2020 scenario in the Gazebo robotic simulator. The interior of the building was updated during the competition to correspond with the interior of the real building, as observed during the rehearsals (see photos from the competition in Figure 11). The Fire Challenge arena is approximately $50\,\mathrm{m} \times 60\,\mathrm{m}$ in dimensions and contains a tall structure (18 m in height) simulating a building. The interior of each floor of the building contains two fire analogues and only one per floor is activated during the trial. Each floor of the building contains eight $2\,\mathrm{m} \times 2\,\mathrm{m}$ windows. Only one of them is open and can be used as an access point to enter
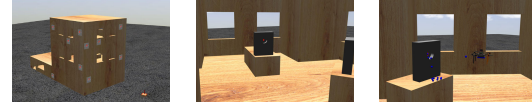
the floor. Snapshots from the simulated scenario are shown in Figure 12.

The behavior of the proposed system can be simulated completely, including the outdoor and indoor flying, window detection, fire detection and also fire extinguishing (see the right image in Figure 12). Numerous simulations were conducted with different settings of the system parameters during the preparations for the competition. The results of the system in the final form after the competition obtained for the evaluation of the system for this paper are shown in Table 1. The goal of each run of the simulation was to extinguish an artificial fire on the first floor of the building. For testing purposes and according to the rules, one of the fire analogues (windows) was randomly selected and was turned on (opened). The position where the MAV started was the same each time for each run of the simulation. Three performance criteria may be considered for an evaluation of the task under discussion in this paper: reliability, total mission time, and minimal distance from the obstacles. The results show that the mission can be completed within 7 min. However, the fire analogue was detected and extinguished only in 80 % of the cases, due to problematic properties of the fire analogues. The fire analogues are visible in the thermal images only under a viewing angle of at most 57° from the position perpendicular to the fire analogue. The proposed exploration method does not consider the angle under which the particular surface in the scene is observed. Therefore with the fire analogues, the system can consider the surface as already scanned even though the fire analogue was not detected, because the surface was scanned under an angle from which the heated element could not be detected. This is a specific property of the fire analogues used in the competition, and it will not prevent successful detection of real fires. During these simulations, it was successfully verified that the MAV did not come closer to the obstacles than 0.7 m, which was the minimum obstacle distance set for the indoor motion planning algorithm.

B. REAL-WORLD VERIFICATION
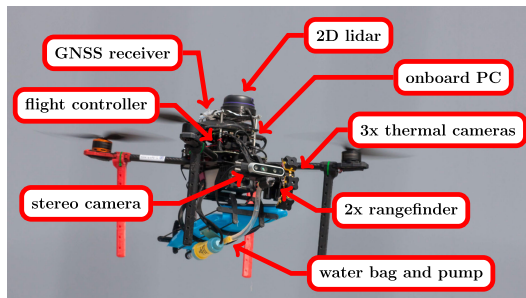#### 1) PLATFORM DESCRIPTION
Our team participated in all challenges of the MBZIRC 2020 competition. To allow reusability of the system and the spare parts, we decided to select a base MAV platform that can be used in all challenges, with possible modifications to the sensors and actuators. The proposed firefighting MAV platform with the complete sensory equipment is shown in Figure 13.

The selected base platform is created mostly from commercially available off-the-shelf components and 3D printed

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

TABLE 1. Table with results from 10 runs of the simulated MBZIRC 2020 competition scenario. The MAV returned back to the starting position each time. However, the fire analogue was detected and was extinguished only in 8 of the cases, due to problematic properties of the fire analogues.

	1	2	3	4	5	6	7	8	9	10
Total time [s]	336.3	368.2	222.3	226.4	218.5	223.1	258.9	165.3	246.9	321.6
Min. distance [m]	0.83	0.73	0.74	0.83	0.78	0.78	0.8	0.87	0.74	0.77
Mission successful?	✓	✗	✗	✓	✓	✓	✓	✓	✓	✓

FIGURE 13. Description of the components of the deployed MAV platform for indoor fire extinguishing.

parts. The platform is built from the *Tarot T650* quadrotor frame, the *PixHawk 4* flight controller,[3] and an *Intel NUC* onboard computer. This frame satisfies the size limitations set for the competition and also for the real applications (the diagonal dimension without propellers is 650 mm), and provides the payload capacity that is necessary for carrying additional sensors and fire-extinguishing equipment. The onboard computer is Intel NUC8i7BEH,[4] which contains Intel i7-8559U CPU and 8 GB of RAM, and runs the Ubuntu 18.04 LTS operating system and Robot Operating System (ROS) [41] Melodic middleware. In addition, the MAV is equipped with *RPLIDAR A3*,[5] which is a 360° 2D laser range scanner that can be used for both indoor and outdoor applications. This sensor provides 16000 samples per second and can detect obstacles within a 25 m radius, depending on the setting of the sensor. For the stereo camera, we use the *Realsense D435* camera,[6] which has FOV (H × V × D) $87°\pm3° \times 58°\pm1° \times 95°\pm3°$ and a range of up to 10 m. Fire detection is done using a set of three *TeraRanger Evo Thermal 33*[7] thermal cameras. This thermal camera is cheap, small, and compact (only 12 g), which is very important in this case of a limited payload. However, the camera has small resolu-

tion of 32 × 32 pixels and FOV of 33° in both dimensions, and requires a set of three of these sensors onboard the MAV to cover the vertical space in front of the MAV sufficiently for this application. The cameras are arranged vertically, with s.t. one pointing forward and the two others above and below it, oriented 30° upwards and downwards from the first camera (see Figure 13). The MAV is further equipped with two *Garmin LIDAR-Lite v3*[8] laser rangefinders.

To extinguish fires, the MAV is equipped with a water bag and a pump.[9] The capacity of the bag was limited to 1 L of the fire-extinguishing agent (water in the case of the competition) to maintain higher maneuverability of the system. This maneuverability is vital for flight in an environment, such as the inside of a building, where strong air currents and various obstacles can be encountered. The pump drives the water through a nozzle with a diameter of 4 mm and can fully deplete the bag within 25 s. The nozzle is rigidly attached to the MAV frame, and is oriented to the front with the spraying tip located 2 cm below and 2 cm in front of it. As has already been mentioned, this nozzle is not actuated, since a servomechanism of this type would significantly increase the weight of the MAV.

2) EXPERIMENTS

The key parts of the proposed system were thoroughly tested in demanding outdoor conditions in the desert near Abu Dhabi in the United Arab Emirates. This environment was selected to emulate the conditions around buildings and other conditions set for the competition itself (mainly sudden wind gusts, strong sunlight and dust), while providing a safe field for system tuning and experimental verification. Repeated experimental verification of the key parts of the proposed system was necessary in order to prepare for phenomena that are difficult to simulate, and also to discover issues related to the hardware that was deployed. One issue that emerged was the influence of sensors connected using USB 3.x., such as *Realsense D435*, on the precision of GNSS. The precision of GNSS localization can be severely decreased by the influence of components transmitting via cable at frequencies close to those used by GNSS. See sheet[10] for a detailed description of USB 3 frequency interference. It was therefore necessary

[3]https://github.com/PX4/px4_user_guide/raw/master/assets/flight_controller/pixhawk4/pixhawk4_technical_data_sheet.pdf

[4]https://www.intel.com/content/dam/support/us/en/documents/mini-pcs/NUC8i3BE_NUC8i5BE_NUC8i7BE_TechProdSpec.pdf

[5]https://www.generationrobots.com/media/LD310_SLAMTEC_rplidar_datasheet_A3M1_v1.0_en.pdf

[6]https://www.intelrealsense.com/wp-content/uploads/2020/06/Intel-RealSense-D400-Series-Datasheet-June-2020.pdf

[7]https://terabee.b-cdn.net/wp-content/uploads/2020/05/evo-thermal_specsheet.pdf

[8]http://static.garmin.com/pumac/LIDAR_Lite_v3_Operation_Manual_and_Technical_Specifications.pdf

[9]https://www.comet-pumpen.de/fileadmin/pdf/pumpen_datenblaetter/24v/Datenblatt_VIP-PLUS_24V_1435.88.00.pdf

[10]https://www.intel.com/content/www/us/en/products/docs/io/universal-serial-bus/usb3-frequency-interference-paper.html
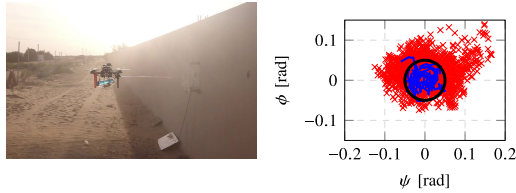
FIGURE 14. Experimental verification of fire extinguishing in the desert near Abu Dhabi, United Arab Emirates. The image on the left shows a photo from this experiment. The image on the right displays the plot of the changing offsets of the nozzle direction from the estimated fire analogue direction during the extinguishing procedure. Angles ϕ and ψ denote vertical (pitch) and horizontal (yaw) angular offsets. The blue line denotes when the water pump was activated. The black circle denotes the range of offsets corresponding to the area of the opening in the plexiglass viewed from the desired extinguishing distance of 1.5 m. Note that most of the time, the water only sprinkled when the nozzle was aiming into the opening. However, the real world dynamics of the water stream caused more spillage than the aim itself implies. A video of this experiment is on YouTube[11].
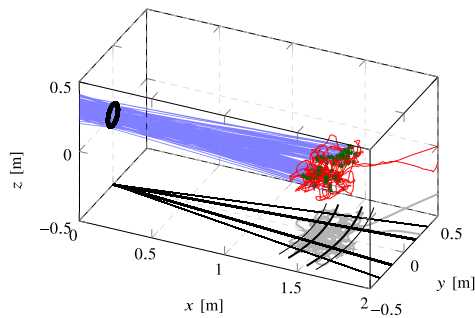

FIGURE 15. Plot of the MAV trajectory during the extinguishing experiment displayed in Figure 14. The red line denotes the trajectory itself relative to the fire analogue. The green points are specific positions from which the MAV activated its water pump. Spraying lines parallel to the direction of the water nozzle in the spraying positions are projected in blue. Note that 93% of these spraying lines pass through the black circle denoting the opening in the plexiglass of the fire analogue. The trajectory is projected to the *xy*-plane as a shadow, for better clarity. Also shown in the *xy*-plane are the hysteresis ranges described in Figure 9.

to shield the receiver of the GNSS signal. Another issue that we discovered was the necessity to calibrate the fire-extinguishing device to hit the fire detected by the thermal cameras precisely for each MAV. Otherwise, the ejected water would not precisely hit the opening in the fire analogues. In fact, the direction of the water stream was diverted downwards by the pressure generated by the active propellers. Although precise placement of the fire-extinguishing agent is also important consideration in real firefighting, the small size of the opening in the MBZIRC 2020 fire analogues presented a much more difficult challenge than a firefighting MAV would face in a real fire.

FIRE DETECTION AND EXTINGUISHING

The first experiment presented here was focused on fire detection in conjunction with autonomous fire extinguishing

(discussed in section II-D and section II-E). The initial goal of this experiment was to detect the fire analogues. After successful detection, the MAV moves in front of the fire at a distance of 1.5 m while heading towards the opening in the center of the fire analogue, and then it initiates the autonomous fire extinguishing. Whenever the MAV points the nozzle towards the opening at the correct relative distance, the water pump is activated (see Figure 14 and Figure 15). The experiment shown in Figure 14 and 15 was carried out in the latest stage of system development prior to the competition, representing the final state of the fire-extinguishing subsystem. A video showing this experiment is available on YouTube.[11] As the data shows, at least in terms of position and heading, the MAV approached the desired extinguishing position w.r.t. the fire analogue and deployed water into the small opening. In this experiment, for 93 % of the time when the water pump was activated, the water nozzle was aiming correctly at the opening. The remaining 7 % was affected by the delay until the pump turned off successfully. This shows the accuracy of the fire detection and localization system. It should be noted that some of the deployed water was lost due to various effects such as dispersion, the momentum of the liquid in the spraying system, surface tension within the water stream, bouncing off from the back plate of the fire analogue, stronger ballistic curvature when the pump is being activated or deactivated, and evaporation from the heating element. Note also that there were numerous interruptions in the correctly-aimed water spraying. These interruptions were caused by loss of the target by the thermal cameras due to the heating element being temporarily cooled down by the deployed water (this was a special property of the MBZIRC 2020 fire analogue, not of a real fire). For this implementation, we decided that it was a better strategy to let the target heat up again and to invest additional time instead of continuously spraying merely based on the "remembered" position of the target. Such estimation without new measurements drifts from the real position, and the limited carrying capacity of an MAV makes it a priority to be economical with the extinguishing agent. With real hazardous fires, such losses of vision will only occur after the extinguishing has been successful, so it would not be necessary to interrupt the water stream.
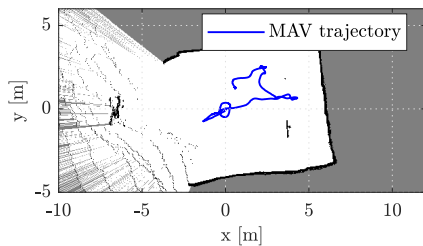
INDOOR MOTION PLANNING AND EXPLORATION

Another experimentally verified subsystem was indoor motion planning and exploration (discussed in section II-C). The goal of the experiment was to completely explore the space inside a room with obstacles consisting of poles holding the structure and wooden artificial obstacles. An example layout of the obstacles inside the room is shown in Figure 16(a). A visualization of data from the experiment in this setup is shown in Figure 16(b). Figure 17 shows a visualization of the MAV trajectory and minimal obstacle distance progression from one of the performed flights. During

[11] https://youtu.be/9bkvfi5uHK4

59

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

(a) Side view from an external camera.

(b) Data visualization.

FIGURE 16. Experimental verification of the indoor motion planning and exploration techniques in the desert near Abu Dhabi, United Arab Emirates before MBZIRC 2020. In the image on the right, the red blocks denote obstacles, the green blocks show the already scanned space, and the yellow blocks denote frontiers for exploration. A video from this experiment is on YouTube[12].

(a) MAV trajectory with a map produced by Hector SLAM is drawn in the background.
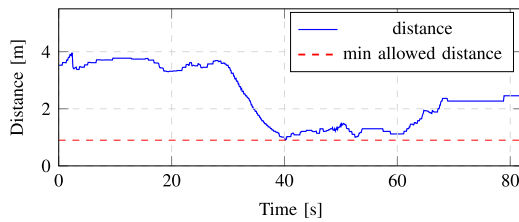
(b) Distance of the MAV from obstacles during the flight.

FIGURE 17. Data from one of the flights from the indoor motion planning and exploration experiment displayed in Figure 16.

the flights, the MAV had not come closer than the specified minimal distance of 0.9 m to the observed obstacles, and had successfully explored the space inside the room. A video showing one of these experiments is available on YouTube.[12]

WINDOW DETECTION AND THE OUTDOOR-INDOOR TRANSITION

Another experiment in the desert was aimed at verifying the correct performance of outdoor wall following, window detection, flight through the detected window, and switching between indoor and outdoor modes of localization. For this purpose, a wooden room was constructed next to a long wall. The room was approximately 2.5 m × 3 m × 2 m in size and contained an entrance 2 m × 2 m in size. The size of the room did not match the MBZIRC 2020 specification but it was suitable for testing these particular parts of the sys-

[12]https://youtu.be/9LTf6PG4ijc
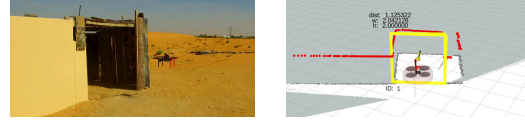
(a) MAV in front of the mock-up building.

(b) Visualization of sensor data.

FIGURE 18. Images from experimental verification of the correct performance of outdoor wall-following, window detection, flight through the detected window, and switching between indoor and outdoor modes of localization. 18(a) shows the MAV before entering a mock-up building in the desert and 18(b) displays a particular visualization of the onboard sensor data. The visualization includes the MAV position, LIDAR data shown as red squares, the detected window as a yellow rectangle, and a map produced by the Hector SLAM algorithm. A video from this experiment is on YouTube[13].

tem. Figure 19(a) displays the trajectory of the MAV during the experiment. The MAV started 2.5 m from the wall and autonomously detected the wall as the closest object seen by the 2D LIDAR, then followed the wall at a distance of 2 m while simultaneously searching for a window. During the outdoor flight, the MAV was localized using GNSS. The window was detected using a combination of 2D LIDAR data and a priori information about its size and altitude. When the window was detected, the MAV approached the window, after which the localization switched to indoor mode (using 2D LIDAR-based Hector SLAM), and the MAV flew inside. The MAV then turned around inside the building, flew back outside, and the localization mode switched back to GNSS. Finally, the MAV returned back to its starting position and landed. The whole experiment, along with a visualization of the sensor data, can be seen in a video on YouTube.[13] Figure 18 shows the MAV in front of the building along with a visualization of the sensor data and the detected window. Figure 19(b) contains a plot of the total control error (defined as the 3D Euclidean distance between the current reference and the MAV position) from the entire flight. This graph shows that the switch between the two different localization systems was smooth and did not impact the control of the MAV. The average control error during the flight was 0.14 m.

COMPLETE SYSTEM VERIFICATION

The complete system was tested in a mock-up of the competition building set up in the Czech Republic. The mock-up is 5 m × 5 m, with 2 floors totaling 5 m in height, with windows 1.85 m × 1 m in dimensions (see Figure 20). The size of the windows matches the first specification for the MBZIRC 2020 competition. This specification was changed later, and the windows are smaller than in the competition. This made the flight through the window more challenging than was necessary for the competition, but it verified the performance and the robustness of the system for real-world deployment. In the experiment presented here, the MAV began next to the mock-up, autonomously detected the wall as the closest object seen by 2D LIDAR, and then followed
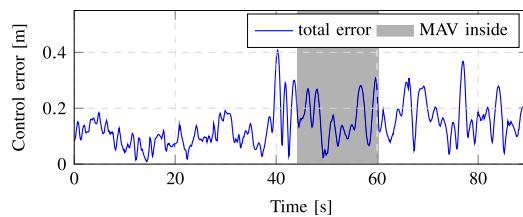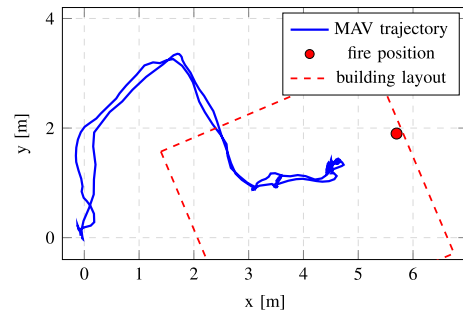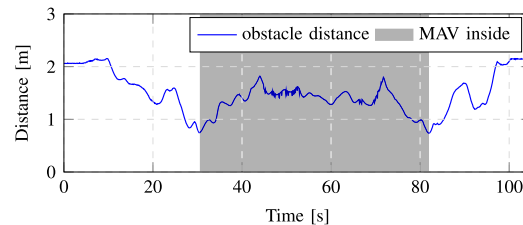
[13]https://youtu.be/aCKUjbJ2Mxs
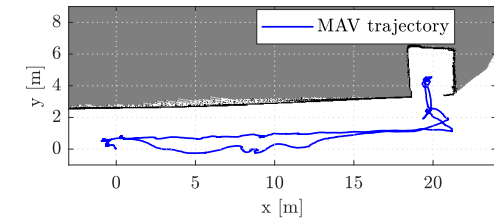
(a) MAV trajectory with a map produced by Hector SLAM drawn in the background.

(b) Control error plot.

FIGURE 19. Data from the outdoor-indoor transition experiment displayed in Figure 18.

(a) Trajectory traveled by the MAV projected to the *xy*-plane.

(b) Closest obstacle distance measured by the 2D LIDAR.

FIGURE 21. Data from the complete system verification experiment presented in Figure 20.

FIGURE 20. Experimental verification of the proposed system on a mock-up of the building set up in the Czech Republic. The image on the left shows the MAV outside the mock-up during its search for an open window. The image on the right displays the MAV while the water is being sprayed on the fire analogue inside the mock-up. A video of this experiment is on YouTube[14].

the wall at a distance of 1.5 m while simultaneously searching for a window. The window was detected using a combination of data from 2D LIDAR and from the stereo camera. After successful detection, the MAV approached the window and flew inside. Then, the MAV started to explore the interior of the building with the goal to find the fire analogue and then to extinguish the fire. The fire analogue was later detected and all the water was depleted on it. Finally, the MAV flew out of the building, using the same window through which had flown in, and then flew back to the starting position of the mission, where it landed. The complete trajectory traveled by the MAV during this experiment is shown in Figure 21(a). Figure 21(b) shows the distance from the closest obstacles measured by 2D LIDAR during the flight. The shortest obstacle distance of the whole flight was 0.74 m, when the MAV was flying through the window. It can be seen that the MAV motion was successfully planned with an adequate safety margin throughout the flight. The whole experiment, along

with a visualization of the sensor data, can be seen in a video on YouTube.[14]

IV. GOING BEYOND THE MBZIRC COMPETITION

An MAV can carry a water bag filled only with a very limited amount of water, due its limited payload capacity. Even after this entire amount has been discharged perfectly into a real fire, the fire will very likely not be extinguished. To increase the payload capability of the MAV, its size has to be changed. However, greater dimensions of the MAV will make flying through windows and inside buildings very difficult to achieve, if not even impossible. Therefore, in the most cases the fire-extinguishing approach, with the water spray, is not the optimal solution.

Based on our results in the competition, the proposed autonomous system was selected to be the core of an industrial firefighting MAV system using fire-extinguishing capsules.[15] This system makes it necessary to hit the fire directly, meaning that it requires reliable techniques for locating, approaching, and aiming precisely at fires. These techniques are being adapted from the work presented here, combined with a throwing mechanism able to place the active fire-fighting capsules quickly and precisely.[16] A prototype of a complex industrial platform is shown in Figure 22.

[14]https://youtu.be/a-VsVQcMLuQ
[15]http://www.fire-defender.com/en/bonpet-3/1465-2
[16]http://mrs.felk.cvut.cz/projects/dofec

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES
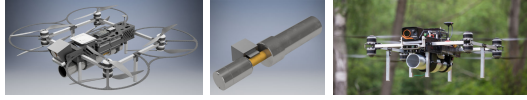
FIGURE 22. A new platform under development that uses fire-extinguishing capsules as projectiles. The first two photos on the right show 3D render of the platform and launcher design, photo on the left shows constructed platform.

V. CONCLUSION

In this work, we have presented a complex system developed for fully autonomous extinguishing of fires inside a building using an MAV system. The challenges include control and estimation of the MAV state, interior motion planning and exploration, window and fire detection and position estimation, and fire extinguishing. One of the main contributions of the system lies in precise control for flying through relatively small windows, and also for precise spraying of fire-extinguishing agent into a small opening representing the fire, using multiple sensory data to increase reliability. For this, we have presented a novel multi-layer control pipeline that further enables precise localization and stabilization in an open space around a building, inside rooms with obstacles, and also with a smooth transition between these two environments (with GNSS and GNSS-denied). This smooth transition is another contribution that motivated the MBZIRC 2020 committee to design this demanding challenge. The paper contains the system performance presented in simulations and field tests in various demanding real-world conditions. The system was developed as part of a solution for the firefighting mission in the MBZIRC 2020 competition, and it helped our team to achieve first place in the Grand Challenge of this competition among the best universities in aerial robotics worldwide.[17] Although the system was developed specifically for this competition, the solution presented here has led to an industrial solution that is now under development. This solution will be focused on real-world firefighting, in which autonomous drones will deploy fire-extinguishing capsules.

REFERENCES

[1] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges," *IEEE Access*, vol. 7, pp. 48572–48634, 2019.

[2] S. Verykokou, C. Ioannidis, G. Athanasiou, N. Doulamis, and A. Amditis, "3D reconstruction of disaster scenes for urban search and rescue," *Multimedia Tools Appl.*, vol. 77, no. 8, pp. 9691–9717, Apr. 2018.

[3] D. Duarte, F. Nex, N. Kerle, and G. Vosselman, "Towards a more efficient detection of earthquake induced facade damages using oblique uav imagery," in *Proc. IEEE ICUAVG*, Aug. 2017, pp. 93–100.

[4] Y. Yamazaki, M. Tamaki, C. Premachandra, C. J. Perera, S. Sumathipala, and B. H. Sudantha, "Victim detection using UAV with on-board voice recognition system," in *Proc. 3rd IEEE Int. Conf. Robotic Comput. (IRC)*, Feb. 2019, pp. 555–559.

[5] M. Rahnemoonfar, R. Murphy, M. V. Miquel, D. Dobbs, and A. Adams, "Flooded area detection from UAV images based on densely connected

[6] R. Ravichandran, D. Ghose, and K. Das, "UAV based survivor search during floods," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2019, pp. 1407–1415.

[7] C. Yuan, Z. Liu, and Y. Zhang, "UAV-based forest fire detection and tracking using image processing techniques," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Jun. 2015, pp. 639–643.

[8] J. Delmerico, S. Mintchev, A. Giusti, B. Gromov, K. Melo, T. Horvat, C. Cadena, M. Hutter, A. Ijspeert, D. Floreano, L. M. Gambardella, R. Siegwart, and D. Scaramuzza, "The current state and future outlook of rescue robotics," *J. Field Robot.*, vol. 36, pp. 1–21, Oct. 2019.

[9] T. Rouček, M. Pecka, P. Čížek, T. Petříček, J. Bayer, V. Šalanský, D. Heřt, M. Petrlík, T. Báča, V. Spurný, F. Pomerleau, V. Kubelka, J. Faigl, K. Zimmermann, M. Saska, T. Svoboda, and T. Krajník, "DARPA subterranean challenge: Multi-robotic exploration of underground environments," in *Proc. IEEE MESAS*, Mar. 2020, pp. 274–290. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-43890-6_22

[10] F. Mascarich, S. Khattak, C. Papachristos, and K. Alexis, "A multi-modal mapping unit for autonomous exploration and mapping of underground tunnels," in *Proc. IEEE Aerosp. Conf.*, Mar. 2018, pp. 1–7.

[11] M. Petrlík, T. Baca, D. Hert, M. Vrba, T. Krajnik, and M. Saska, "A robust UAV system for operations in a constrained environment," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2169–2176, Apr. 2020.

[12] L. Shi, X. Wang, T. Zhang, C. Hu, K. Luo, and B. Bai, "Hazardous gas detection four-rotor UAV system development," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, Aug. 2016, pp. 2461–2465.

[13] Z. Fu, Y. Chen, Y. Ding, and D. He, "Pollution source localization based on multi-UAV cooperative communication," *IEEE Access*, vol. 7, pp. 29304–29312, 2019.

[14] P. Stibinger, T. Baca, and M. Saska, "Localization of ionizing radiation sources by cooperating micro aerial vehicles with pixel detectors in real-time," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3634–3641, Apr. 2020.

[15] C. Corrado and K. Panetta, "Data fusion and unmanned aerial vehicles (UAVs) for first responders," in *Proc. IEEE Int. Symp. Technol. Homeland Secur. (HST)*, Apr. 2017, pp. 1–6.

[16] L. Merino, F. Caballero, J. R. Martínez-de-Dios, I. Maza, and A. Ollero, "An unmanned aircraft system for automatic forest fire monitoring and measurement," *J. Intell. Robotic Syst.*, vol. 65, nos. 1–4, pp. 533–548, Jan. 2012.

[17] A. Restas, "Forest fire management supporting by UAV based air reconnaissance results of Szendro fire department, Hungary," in *Proc. 1st Int. Symp. Environ. Identities Medit. Area*, Jul. 2006, pp. 73–77.

[18] A. Viguria, I. Maza, and A. Ollero, "Distributed service-based cooperation in aerial/ground robot teams applied to fire detection and extinguishing missions," *Adv. Robot.*, vol. 24, nos. 1–2, pp. 1–23, Jan. 2010.

[19] R. Chen, H. Cao, H. Cheng, and J. Xie, "Study on urban emergency firefighting flying robots based on UAV," in *Proc. IEEE 4th Adv. Inf. Technol., Electron. Autom. Control Conf. (IAEAC)*, Dec. 2019, pp. 1890–1893.

[20] H. Qin, J. Q. Cui, J. Li, Y. Bi, M. Lan, M. Shan, W. Liu, K. Wang, F. Lin, Y. F. Zhang, and B. M. Chen, "Design and implementation of an unmanned aerial vehicle for autonomous firefighting missions," in *Proc. 12th IEEE Int. Conf. Control Autom. (ICCA)*, Jun. 2016, pp. 62–67.

[21] P. Pecho, P. Magdolenová, and M. Bugaj, "Unmanned aerial vehicle technology in the process of early fire localization of buildings," *Transp. Res. Procedia*, vol. 40, pp. 461–468, Jan. 2019.

[22] A. Imdoukh, A. Shaker, A. Al-Toukhy, D. Kablaoui, and M. El-Abd, "Semi-autonomous indoor firefighting UAV," in *Proc. 18th Int. Conf. Adv. Robot. (ICAR)*, Jul. 2017, pp. 310–315.

[23] M. Popp, S. Prophet, G. Scholz, and G. F. Trommer, "A novel guidance and navigation system for MAVs capable of autonomous collision-free entering of buildings," *Gyroscopy Navigat.*, vol. 6, no. 3, pp. 157–165, Jul. 2015.

[24] M. Popp, G. Scholz, S. Prophet, and G. F. Trommer, "A laser and image based navigation and guidance system for autonomous outdoor-indoor transition flights of MAVs," in *Proc. DGON Inertial Sensors Syst. Symp. (ISS)*, Sep. 2015, pp. 1–18.

[25] S. Zhou, G. Flores, E. Bazan, R. Lozano, and A. Rodriguez, "Real-time object detection and pose estimation using stereo vision. An application for a quadrotor MAV," in *Proc. Workshop Res., Edu. Develop. Unmanned Aerial Syst. (RED-UAS)*, Nov. 2015, pp. 72–77.

[26] G. Flores, S. Zhou, R. Lozano, and P. Castillo, "A vision and GPS-based real-time trajectory planning for MAV in unknown urban environments," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, May 2013, pp. 1150–1155.

[17] https://www.mbzirc.com/winning-teams/2020

[27] R. C. Julian, C. J. Rose, H. Hu, and R. S. Fearing, "Cooperative control and modeling for narrow passage traversal with an ornithopter MAV and lightweight ground station," in *Proc. Citeseer AAMAS*, 2013, pp. 103–110.

[28] G. Loianno, C. Brunner, G. McGrath, and V. Kumar, "Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and IMU," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 404–411, Apr. 2017.

[29] D. Falanga, E. Mueggler, M. Faessler, and D. Scaramuzza, "Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 5774–5781.

[30] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. Grixa, F. Ruess, M. Suppa, and D. Burschka, "Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue," *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 46–56, Sep. 2012.

[31] K. Schmid, P. Lutz, T. Tomić, E. Mair, and H. Hirschmüller, "Autonomous vision-based micro air vehicle for indoor and outdoor navigation," *J. Field Robot.*, vol. 31, no. 4, pp. 537–570, Jul. 2014.

[32] 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," 2020, *arXiv:2008.08050*. [Online]. Available: http://arxiv.org/abs/2008.08050

[33] T. Baca, D. Hert, G. Loianno, M. Saska, and V. Kumar, "Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1–8.

[34] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," in *Proc. 49th IEEE Conf. Decis. Control (CDC)*, Dec. 2010, pp. 5420–5425.

[35] S. Kohlbrecher, O. von Stryk, J. Meyer, and U. Klingauf, "A flexible and scalable SLAM system with full 3D motion estimation," in *Proc. IEEE Int. Symp. Saf., Secur., Rescue Robot.*, Nov. 2011, pp. 155–160.

[36] A. Siadat, A. Kaske, S. Klausmann, M. Dufaut, and R. Husson, "An optimized segmentation method for a 2D laser-scanner applied to mobile robot navigation," *IFAC Proc. Volumes*, vol. 30, no. 7, pp. 149–154, Jun. 1997.

[37] G. Welch and G. Bishop, "An introduction to the Kalman filter," in *Proc. SIGGRAPH*, 2006, p. 16.

[38] V. Usenko, L. von Stumberg, A. Pangercic, and D. Cremers, "Real-time trajectory replanning for MAVs using uniform B-splines and a 3D circular buffer," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 215–222.

[39] W. Minkina and S. Dudzik, *Infrared Thermography: Errors and Uncertainties*. Hoboken, NJ, USA: Wiley, 2009.

[40] P. Schillinger, S. Kohlbrecher, and O. von Stryk, "Human-robot collaborative high-level control with application to rescue robotics," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 2796–2802.

[41] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: An open-source robot operating system," in *Proc. Workshop Open Source Softw. (ICRA)*, vol. 3, 2009, p. 5.
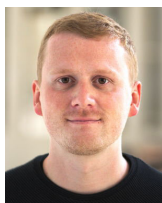
VACLAV PRITZL received the M.Sc. degree from Czech Technical University in Prague, Czech Republic, where he is currently pursuing the Ph.D. degree focused on data fusion in aerial robotics. He focuses on methods enabling indoor exploration for unmanned aerial vehicles. Since 2017, he has been a part of the Multi-Robot Systems Laboratory with CTU Prague. He was a member of the CTU-UPENN-NYU team in the MBZIRC 2020 competition.
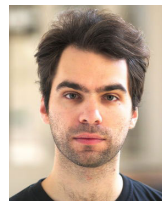
VIKTOR WALTER received the M.Sc. degree from the Brno University of Technology, Czech Republic, where he is currently pursuing the Ph.D. degree in mutual relative localization of unmanned aerial vehicles. Since 2016, he has been a part of the Multi-Robot Systems Laboratory with CTU Prague. He is primarily focused on onboard computer vision for unmanned aerial vehicles and behaviors based on such inputs. He was also a member of the CTU-UPENN-NYU team in the MBZIRC 2020 competition.

MATEJ PETRLIK received the M.Sc. degree from Czech Technical University in Prague, Czech Republic, where he is currently pursuing the Ph.D. in sensor fusion, cooperative mapping, and dynamic state estimation of unmanned aerial vehicles. He was also a member of CTU-UPenn-UoL and CTU-UPENN-NYU teams in the MBZIRC 2017 and 2020, and the CTU-CRAS-NORLAB team in the DARPA SubT competition.

VOJTECH SPURNY received the M.Sc. degree from Czech Technical University in Prague, Czech Republic, where he is currently pursuing the Ph.D. degree in methods of planning and coordination for unmanned aerial vehicles. Since 2014, he has been a part of the Multi-Robot Systems Laboratory with CTU Prague. He was a Key Member of CTU-UPenn-UoL and CTU-UPENN-NYU teams in the MBZIRC 2017 and 2020 robotic competitions in Abu Dhabi, United Arab Emirates.

TOMAS BACA (Graduate Student Member, IEEE) received the M.Sc. degree from Czech Technical University in Prague, Czech Republic, where he is currently pursuing the Ph.D. degree in distributed remote sensing. Since 2014, he has been a part of the Multi-Robot Systems Laboratory with CTU Prague. He focuses on distributed radiation sensing with unmanned aerial vehicles and planning and control for unmanned aerial vehicles. He was a Key Member of CTU-UPenn-UoL and CTU-UPENN-NYU teams in the MBZIRC 2017 and 2020 robotic competitions in Abu Dhabi.

Final version is available at `https://ieeexplore.ieee.org/document/9476723`

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

PETR STEPAN received the M.Sc. degree from Charles University, Czech Republic, in 1994, and the Ph.D. degree from Czech Technical University in Prague, Czech Republic. Since 1996, he has been a Research Fellow with Czech Technical University in Prague. He focuses on sensor fusion, mapping, and localization tasks. He participates in six EU project about autonomous robot. He was also a member of CTU-UPenn-UoL and CTU-UPENN-NYU teams in the MBZIRC 2017 and 2020.

MARTIN SASKA (Member, IEEE) received the M.Sc. degree from Czech Technical University in Prague, in 2005, and the Ph.D. degree from the University of Wuerzburg, Germany. Since 2009, he has been a Research Fellow with Czech Technical University in Prague, where he has founded and since leads the Multi-Robot Systems Laboratory and co-founded the Center for Robotics and Autonomous Systems. He was a Visiting Scholar with the University of Illinois at Urbana-Champaign, Champaign, IL, USA, in 2008, and the University of Pennsylvania, Philadelphia, PA, USA, in 2012, 2014, and 2016. He is the author or a coauthor of more than 70 publications in peer-reviewed conferences and more than 20 publications in impacted journals.

DAVID ZAITLIK received the M.Sc. degree from Czech Technical University in Prague, Czech Republic. He is currently a Research Fellow focused on embedded systems. Since 2019, he has been a part of the Multi-Robot Systems Laboratory with CTU Prague. He was a member of the CTU-UPENN-NYU team in the MBZIRC 2020 competition.

...

64

Systems Article

Extinguishing Real Fires by Fully Autonomous Multirotor UAVs in the *MBZIRC 2020* Competition

Viktor Walter[1] , **Vojtěch Spurný**[1] , **Matěj Petrlík**[1] , **Tomáš Báča**[1] , **David Žaitlík**[1] ,
Lyubomyr Demkiv[2] **and Martin Saska**[1]

[1]Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague,
http://mrs.felk.cvut.cz

[2]Robotics Group of SoftServe Inc. and with the Institute of Computer Science and Information Technologies,
Lviv Polytechnic National University, Ukraine

Abstract: In this paper, we describe a system for combating real fires with sprayed liquid extinguishing agent using a team of multirotor UAVs. The system design relies on onboard sensors and operates without the need for human intervention throughout its complex mission, from its takeoff to landing. The core UAV platform can estimate its state, self-localize, navigate and locate and extinguishing fires. Thermal and RGB cameras are used, each with a specialized computer vision subsystem and are combined with planar LIDAR for 3D localization of fires on multistory building facades. The system conducts aerial firefighting with a software stack that addresses flight dynamics and sensor limitations and a liquid-spraying subsystem incorporating a rigidly-attached water nozzle. The approach presented in this paper was motivated by the Mohamed Bin Zayed International Robotics Challenge (MBZIRC 2020) firefighting scenario, which focused on coordinated multi-UAV teams that can autonomously combat high-rise building fires. The MBZIRC series places particular emphasis on fast and reliable deployment of robots in realistic conditions. These contests promote development of real-world applications that are greatly needed by society, but which still exceed State-of-the-Art in the robotics community. To our knowledge, our system was the only MBZIRC 2020 contender to extinguish a facade fire successfully in autonomous mode without using an RTK-GNSS system. Our approach contributed to victory in the overall competition and we have now adapted it into an industrial prototype for a firefighting UAV. A video attachment to this paper is available at http://mrs.felk.cvut.cz/fr2020firechallenge-facadefires.

Keywords: emergency response, aerial robotics

1. Introduction

There is currently an increasing demand for Unmanned Aerial Vehicles (UAVs) in the disaster-relief and emergency-response sectors. In 2017 alone, worldwide fire casualties reached 120,632 people

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

(Ritchie, 2018) and death from fire or heat was the 4th leading cause of death by accidental injury in 2019, with over 110,000 victims (WHO, 2020). Additionally, there were 1,318,500 fires in the United States during 2018, causing 3,655 deaths and 15,200 injuries. The total losses from fires in 2018 were equal to $25.6 billion US dollars (USFA, 2018; NFPA, 2019).

Robots present one possible solution to helping firefighters and decreasing the amount of fire-related casualties. The robots can aid firefighters in detecting and locating fires within buildings and extinguishing them, both indoors and outdoors (Safetymanagement, 2018). Such systems can also provide firefighters with the information needed to protect the lives of firefighters during incidents. For example, among the robots used in disaster response is

- THOR (Tactical Hazardous Operations Robot), a humanoid robot designed by Virginia Tech University and employed by the U.S. Navy's Shipboard Autonomous Firefighting Robot (SAFFiR) program. Its goal is to assist sailors in firefighting onboard ships. Unfortunately, this robot is slow and may be vulnerable to water damage (VT, 2018; Hopkins et al., 2015).
- Thermite Robot is a system created by Howe and Howe Technologies for the U.S. Army. It is a remote-controlled mobile robot equipped with a hose capable of delivering 1900 liters of water per minute (Firehouse, 2018).
- TAF 20 and TAF 35 (Turbine Aided Firefighting Machine) by Emicontrols (a subsidiary of the TechnoAlpin Group) operates well in small spaces, such as tunnels. The robot can move obstacles with its bulldozer blade and clear the environment of smoke and gasses by binding them within its turbine. It is also able to discharge 3,500 liters of water per minute (Emicontrols, 2018a; Emicontrols, 2018b).
- TC800-FF by Tecdrone is a remotely operated robot capable of working indoors and outdoors. It is equipped with multiple sensors to retrieve data from disaster-stricken areas and to then send it back to remote operators (Robotpompier, 2018).
- Fire OX (Fire Product Search, 2018) by Lockheed Martin is capable of carrying a water tank, standing out among other disaster robots due to its capability of semi-autonomous operation.

These are merely samples of existing solutions. Other firefighting robotic systems are also in use (Lattimer, 2020; Science|Business, 2020; Peskoe-Yang, 2019).

While these robots share a common need for remote operation, they are also unsuitable for combatting fire on tall buildings. Though they can ascend stairs, when speed is essential these systems are outperformed by UAVs. The robots used during the fire at Notre Dame Cathedral (Science|Business, 2020; Peskoe-Yang, 2019) could not help firefighters abate the fire, but rather helped them fight with assessing the consequences and to save priceless artefacts. However, even though UAVs were used as well, they were only applied to monitor the scene and to select locations for deploying hoses by humans. Thermal cameras, which would have have been highly practical for the scenario, were not used (The Verge, 2019).

Moreover, operating within environments with low visibility makes it harder for a remote operator to control the robot. Such limitations increase the importance of autonomous robots in navigating smoky or hazardous environments, since they are able to navigate using data from sensors of diverse modalities, such as rangefinders, thermal and visual cameras, etc.

Fires on high-rise building facades pose a particularly relevant challenge and opportunity for robots. An emergency evacuation in such a building can be difficult and may thus lead to injuries or the loss of human life and since such fires are out of reach for conventional, ground-based firefighting methods, an aerial solution is called for. The timing aspect of this problem is the decisive factor for the amount of damage the fire will cause. According to recent analyses, the incipient stages of fires on the cladding materials of modern high-rise buildings take several minutes, during which the size of the fire is relatively manageable. After this phase, the fire rapidly spreads beyond the control of firefighting efforts and considerable damages are then unavoidable (Chen et al., 2019b; Guillaume et al., 2018; McKenna et al., 2018). This observation is consistent with the timeline of the fire in the Grenfell Tower in June 2017 (London Fire Brigade, 2018). One strategic option would

3.2. ARTICLE 5: AERIAL LIQUID-BASED REAL FIRE EXTINGUISHING

408 · Walter et al.

maintain a team of UAVs with firefighting capabilities on-site or ready for delivery by the local first responders deployed by local fire departments. Such systems could be deployed significantly sooner than a manned helicopter mission, which might follow later.

In order to fully exploit the potential of UAVs in firefighting, especially in high-rise buildings, and to achieve reliable operation, the UAVs themselves need to be autonomous with capabilities going beyond merely following a static, predefined trajectory. The precise location of a fire source, especially in hard-to-reach locations such as high-rise buildings, is difficult to specify accurately from a takeoff location. The fire source may not even be directly visible from the operator's vantage point, apart from smoke plumes indicating an approximate region of interest.

On the other hand, the human factor in UAV-based firefighting can be indispensable. For example, with vision-based fire localization onboard a UAV, finding fire sources without at least an approximate prior estimate of the region of interest can be time-consuming. Human oversight can specify, for example, the range of floors or the building face where the fire is most likely located, based on external observation, sensors inside the building, and other information sources available to the human operators. This input can then be used by the UAV system to navigate quickly and autonomously to a location where onboard sensors can identify and locate the fire sources. While merely flying close to a point outside of a building in an otherwise open space is simple to execute, it is time-consuming if the distance is great. For this reason, manual control of the UAV in the approach phase would waste human time and attention that could better be applied to other critical aspects of the entire firefighting mission.

Autonomy is additionally indispensable in the procedure of extinguishing the fires themselves. Since direct mechanical contact with the environment by UAVs is hazardous, the fires may have to be doused with a liquid stream or fire-extinguishing projectiles. For such scenarios a fire-fighting UAV must be able to aim at the localized fire sources. The ideal position at which the extinguishing agent should be deployed at any given moment will change with time and with targets being affected by extinguishing. Controlling this operation manually is unfeasible in conditions typical of the exteriors of higher floors of high-rise buildings. In such locations, the UAV and the fire source can be too far away from ground-based operators for them to effectively take aim, even if the UAV system features a low-latency onboard video streaming system. Wind often present in such conditions would also make manually correcting aim highly challenging, as had been observed during the Mohamed Bin Zayed International Robotics Challenge (MBZIRC 2020) Challenge 3 where some teams employed teleoperated solutions without success.

Saving endangered people and property is an often-cited motivation for the development of new robotic systems. However, this factor is rarely as immediately applicable as it was in Challenge 3 of the MBZIRC 2020 competition which was directly inspired by the need to extinguish fires in difficult to access locations, such as the high-rise buildings discussed above. This challenge required the use of a team of firefighting robots to locate and extinguish fires, both within and without a building. Solving the challenge required multiple robotic tasks to be addressed beyond just hardware design, including navigation, searching for objects, computer vision with 3D localization, sensor fusion, and implementation of active extinguishing procedures.

This paper describes our approach to one specific subtask of the challenge: extinguishing real fires on the facade of a building with water deployed by a multirotor UAV. The approach we employed was based on using thermal cameras combined with a specialized thermal vision and estimation subsystem, as well as custom-designed searching and in-flight extinguishing maneuvers. To the best of our knowledge, we were the only team in this subtask to not only successfully use full autonomy during the flight, but to also rely solely on local sensors for localization of the targets and control of the UAVs.

The main contributions of this paper include the overall system design, as a model for realistic UAV-based firefighting, and the observations made during iterative development and deployment stages. Novel aspects of our system include the onboard, hybrid, vision-based localization of building-facade fires and fire-extinguishing capability, using dynamic body-pose-based aiming and water-jet control. Among the lessons learned in deployment, we emphasize the need for robust computer

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

vision, specifically robust with respect to the highly dynamic appearance of fires in RGB and thermal images. Other lessons include the possibility of exploiting the transparency of flames to blue light in order to observe surfaces behind the flames. There is also a need for robust aiming and localization suitable even for a maneuvering multirotor UAV. More specific technical aspects are also included, such as addressing the severe interference of USB 3.x devices with Global Navigation Satellite System (GNSS) equipment.

2. Related Work

Using UAVs in firefighting has been previously investigated in several works. One situation where UAVs can help is in monitoring of outdoor fires. As reported in (Merino et al., 2012), a system of multiple UAVs can be used for automatic forest fire monitoring using RGB and infrared cameras. Real experiments with forest fire monitoring in a national park have already been conducted by the Hungarian fire department (Restas, 2006). The authors of (Viguria et al., 2010) describe a task allocation strategy for the distributed cooperation of ground and aerial robot teams in fire detection and extinguishing. In (Chen et al., 2019a), a UAV system is designed to extinguish a fire by dropping a fire-extinguishing capsule on it. Similarly, in (Saikin et al., 2020) the authors employ a UAV system for delivering a liquid fire-extinguishing agent by a dynamic dropping maneuver, maximizing release velocity to reduce dispersion.

However, none of the systems above address certain aspects, such as environmental interaction, precision control for extinguishing, or flight close to buildings (much less flight inside them). In order to solve these tasks autonomously, the following challenges need to be addressed: Precise localization near obstacles with minimal reliance on GNSS, autonomous real-time detection of fires using onboard sensors, reliable stabilization of UAV during extinguishing, motion planning, and reactive obstacle avoidance.

Multirotor UAVs, also known as drones, are widely used in multiple areas of industry and the public sector, as is evident in surveys (Shakhatreh et al., 2019). For instance, the CB Insight report (CBInsight, 2020) lists 38 technology sectors where UAVs are highly applicable were listed, among which were education, space, defense, disaster relief, etc. An extensive survey of the application of drones for bridge inspection is performed in (Hubbard and Hubbard, 2020). While the methodology presented in the paper establishes a basic framework intended for bridge inspection, the approach can be tailored for inspections in other domains, such as the firefighting mission under consideration. For the inspection, a mission definition system must be set up. Such multirotor UAV-oriented architecture is presented in (Portas et al., 2018). Another mission guided framework was presented in (Hrabia et al., 2019). The investigation of automated inspection applications is conducted in (Tudevdagva et al., 2017) for high voltage transmission lines, and in (Ashour et al., 2016) for inspecting and regulating construction sites. As mentioned in the Deloitte analysis report (Schatsky and Ream, 2016), UAVs can increase the speed of wind turbine inspection up to 10 times in comparison to traditional methods. In the report published by (Measure, 2016), benefits and use cases of UAVs in power generation facilities are also analyzed. However, these systems are primarily intended only to assist in gathering information for human operators using manual control and telemetry.

During a firefighting mission, one of the objectives is to detect specific objects, typically flames, and to determine their coordinates. This can be done in multiple ways. One of the methods that is becoming increasingly common in practical use is computer vision. Images provide a rich information source on a relatively large section of the environments, but they require specialized processing to be utilized in autonomous systems. A survey regarding the computer vision methods and corresponding data-sets for UAV flying at different altitudes (e.g., eye-level, low-medium, aerial) is discussed in (Cazzato et al., 2020). Among the most popular ways is the utilization of the machine learning (ML) approach. For instance in (Nousi et al., 2019), a Robot Operating System (ROS)-based environment was used to investigate the applicability of various approaches (executed on Nvidia Jetson TX2) for real-time object detection and tracking. The results obtained in this paper testify

410 · Walter et al.

to the possibility of running Neural Network (NN)-based detection on an embedded device. While such ML methods are capable of exceptional object recognition capabilities in certain situations, they present numerous challenges. Besides the considerable onboard computational power necessary for such a hazardous application, the highly varied appearance of fires and the unpredictable backgrounds and lighting in real-world situations can make it unfeasible to obtain sufficiently representative training datasets.

In general, for projects where the goal is to detect fire, a vision based approach using classical color imaging alone may not be enough as the fire itself may be obscured from cameras by various solid surfaces or smoke. For this purpose, in (Pecho et al., 2019), a thermal camera for indoor building inspection was used. The thermal imaging proved to be a good source of data on the interior fires, but the UAV was only piloted manually, with the thermal images being transmitted for appraisal by human operators instead of being processed automatically. Thermal cameras are also useful for the detection of other objects beside fires or when, for example, detection is challenging due to light conditions in the visible spectrum (Andraši et al., 2017). Simultaneous utilisation of both thermal and RGB cameras is also a common form of sensor fusion. For instance in (Vidas et al., 2013), such a sensor combination is used for the construction of a dense 3D model of an investigated object. It is used for crowd monitoring in (Schulte et al., 2017) and for small UAV detection in (Goecks et al., 2020).

The robotic extinguishing of fire with a water stream has been previously explored, but primarily with ground robots (Liljeback et al., 2006; Rangan et al., 2013) that are mechanically stabilized by contact with the ground. The kinematics of a water stream deployed from a UAV were explored in (Lee et al., 2020) and aiming with UAV body position was explored in (Vrba et al., 2019). However, these works were not focused on correctly aiming for extended periods of time in the context of firefighting. Firefighting using UAVs has already also been a topic of other robotics competitions. (Qin et al., 2016) describes the design and implementation of a firefighting UAV for outdoor applications designed specifically for the IMAV 2015 competition.

3. MBZIRC 2020 Challenge 3

This paper is mainly motivated by the MBZIRC 2020 competition. The competition was divided into three challenges plus a Grand Challenge and final Exhibition, all of which our team participated in, each motivated by a different set of real-world problems. Challenge 3 was focused on robotic firefighting in various conditions and this paper discusses our approach to its first subtask, specifically extinguishing real fires that were emitted by a gas pipe on the facade of a building with a water stream.

3.1. Challenge subtasks

Challenge 3 of the MBZIRC 2020 comprised four distinct parts: extinguishing real fires on the exterior of a building by UAV with water, extinguishing fire analogues in the interior of a building by UAV with water, extinguishing fire analogues in the interior of a building by Unmanned Ground Vehicle (UGV) with water, and extinguishing free-standing fire analogues by UAV with fire blankets.

Each of the three challenges progressed as follows: first, each participating team was given three rehearsal runs on the dates of the 20th, 21st, and 22nd of February, 2020. These rehearsals took 15 min each for Challenge 3. Afterwards, two scored attempts were performed as part of the competition proper on the 23rd and 24th of February, 2020, once again for 15 min each. Each challenge of MBZIRC 2020 had a separate ranking based on points achieved with manual control. The teams scoring no points on autonomous control were always ranked below the lowest scoring team with at least some degree of autonomy, reflecting the significant difference in relevance to the field of robotics and the difficulty of execution. The final result of a team was selected as the better result of two attempts. Lastly, the Grand Challenge phase of the competition took place on the
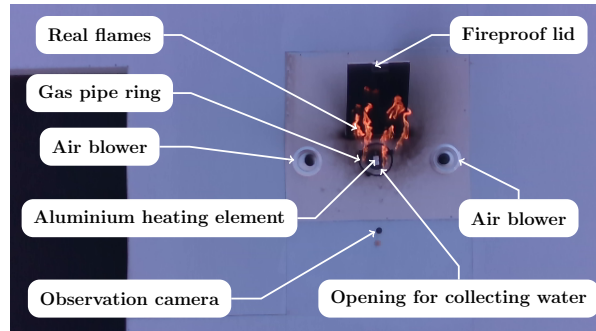
Figure 1. Description of the fire objects used in the competition.

25th of February, where parts of two other challenges (aerial grasping and wall building) were to be performed simultaneously with Challenge 3.

This model of competition enabled a fair comparison of diverse, competing solutions. Each team was afforded an equal number of public attempts and limited rehearsal time. This makes the results more trustworthy than if they were potentially cherry-picked from laboratory testing. Nevertheless, such a small number of attempts makes it impossible to obtain statistically significant test results on the performance of the systems. To address this observation, we complement these results with statistical evaluation of the precision of our aiming procedure.

3.2. Facade firefighting

The Challenge 3 subtask of extinguishing facade fires took place around the facade of a purposefully-built structure resembling a three story building of realistic proportions (referred to hereby as "building"). The real fires (henceforth collectively referred to as "fire objects") were housed in structures located on the facade of the building comprised of multiple parts (see Figure 1):

- A circular ring of perforated metal piping producing ignited gasses — approximately 25 cm diameter
- A circular opening in the facade with a dark interior, containing a system for measurement for the volume of collected water - approximately 15 cm diameter
- An anodized aluminium heating element in the center of the opening — rectangle of 6×3.5 cm
- Air blowers on two sides of the ring active on the ground floor of the building.

These objects were spread out on 3 of the 4 sides of the building, but only one fire object was activated per floor during each run of the subtask. The rest were inactive, meaning that the fire was out and the entire object was covered by an opaque lid of the same color as the building walls. The lid was also closed on the active objects once 1 L of water had been collected through the opening. One of the objects had air blowers activated on the sides, complicating stabilization of the UAV in front of it.

While the specification of the competition task corresponds to a simplified situation compared to a real firefighting mission, it provides a well-modelled representation for such real-life missions. This is because a successful solution in the competition must contain, in some form, all the elements that would otherwise be involved in a real, autonomous, aerial, firefighting system, namely self-localization, navigation, fire source localization, and extinguishing capabilities.

For the liquid-based fire extinguishing subtasks of Challenge 3, the scoring scheme was based on the weighted sum of the amount of water deposited into each specific target. Up to 1 L of water
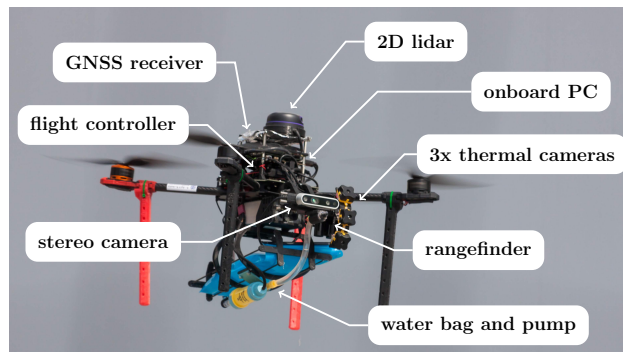
Figure 2. The description of components of the deployed UAV platform for liquid-based fire extinguishing.

could be deposited into each of the six targets. Relevant to the subtask described in this section, the fire objects on the upper floors of the building facade counted as 8 points each multiplied by the amount of deposited water in liters. The total maximum possible score in Challenge 3 of all subtasks was 100 points.

4. Platform Description

Our team participated in all challenges of the MBZIRC 2020 competition. To allow for re-usability of the system and its spare parts, our intention was to select a base UAV platform that could be used in all challenges with the possibility to modify the sensors and actuators. The UAV platform with complete sensory equipment for the task solved in this paper can be seen in Figure 2. The selected base platform is created mostly from commercially available off-the-shelf components and 3D printed parts. The platform is built from the *Tarot T650* quadrotor frame, the *PixHawk 4* flight controller[1], and an onboard computer. This frame satisfies the size limitations for the competition - diagonal dimension without propellers is 650 mm, with the 15 inch propellers attached expanding it to maximum diagonal span of 995 mm, or aligned with the front-back axis making the highest dimension 800 mm. The height of the fully equipped UAV is 410 mm. The payload capacity is sufficient for carrying additional sensors and fire extinguishing equipment. The entire unit weighs 4.1 kg, with the full water bag mentioned below increasing this weight by up to additional 1 kg. The onboard computer is Intel NUC8i7BEH[2] that contains Intel i7-8559U CPU and 8 GB of RAM, and runs the Ubuntu 18.04 LTS operating system and ROS(Quigley et al., 2009) Melodic middleware. For general localization, the UAV carries GNSS receiver based on the *Ublox Neo-M8N* module. Furthermore, the UAV is equipped with the *RPLIDAR A3*[3], which is a 360° 2D LIDAR. This sensor provides 1,600 samples per second and can detect obstacles up to a 25 m radius depending on the setting of the sensor. We also use the *Realsense D435* camera[4] with field of view (FoV) (H × V × D) 87°±3° × 58°±1° × 95°±3° and a range of up to 10 m, but for the facade fire detection, it

---

[1] https://github.com/PX4/px4__user__guide/raw/master/assets/flight__controller/pixhawk4/pixhawk4__technical__data__sheet.pdf - Accessed: 03-30-2022
[2] https://www.intel.com/content/dam/support/us/en/documents/mini-pcs/NUC8i3BE__NUC8i5BE__NUC8i7BE__TechProdSpec.pdf - Accessed: 03-30-2022
[3] https://www.generationrobots.com/media/LD310__SLAMTEC__rplidar__datasheet__A3M1__v1.0__en.pdf - Accessed: 03-30-2022
[4] https://www.intelrealsense.com/wp-content/uploads/2020/06/Intel-RealSense-D400-Series-Datasheet-June-2020.pdf - Accessed: 03-30-2022

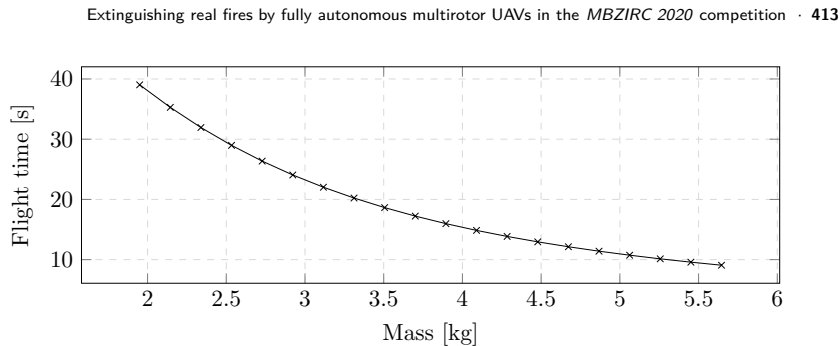Extinguishing real fires by fully autonomous multirotor UAVs in the *MBZIRC 2020* competition · **413**

Figure 3. Empirical evaluation of the flight time of our UAV platform in relation to its total mass including all equipment.[5]

is only used as an RGB camera for precise target localization from close distances. Detection of the fires from greater distance is done using a set of three thermal cameras – *TeraRanger Evo Thermal 33*[6]. This thermal camera is cheap, small, and lightweight (only 12 g) which is very important for a limited payload, as is the case here. The cameras are arranged vertically with one pointing forwards and the two others above and below it with an orientation at 30° upwards and downwards respectively, from the first thermal camera - see Figure 2. The UAV is further equipped with down facing *Garmin LIDAR-Lite v3*[7] laser rangefinder used as a precise altimeter.

To extinguish fires, the UAV is equipped with a water bag and a pump. The capacity of the bag was limited to 1 L of water to maintain high maneuverability of the system. This maneuverability is vital for flight in an environment with possible strong air currents close to hazards. The pump drives the water through a nozzle with a diameter of 4 mm and can fully deplete the bag in 25 s. The nozzle is rigidly attached to the UAV frame and is oriented towards the front with the spraying tip located 2 cm below and 2 cm in front of the *Realsense* camera. Aiming is done using the motion of the UAV itself.

Early in development, we had considered the option of having the nozzle independently actuated to allow for aiming uncoupled from the motion of the UAV. However for the design of the small-scale UAV for the purposes of the competition, we avoided this approach due to various development considerations and limitations. Among other things, these included the increased likelihood of unpredictable behavior in untested states brought about by added mechanical and subsequent software complexity in the system, added weight to UAV with limited carrying capacity, and precision of the available actuators. Additionally, our experience afforded us the expectation that the repair work of UAVs damaged in testing and the competition runs would be significantly more difficult and costly.

It should be noted that while water was used, because of the specifications of the competition, the platform could easily have been adapted to other liquid fire extinguishing agents, as would be demanded by real-world conditions of use.

The use of an extinguishing agent with a better weight to effect ratio would be especially beneficial when the flight time of UAV is considered. The batteries used in the competition on our platforms had the capacity of 177.6 W h, allowing for 10 min of flight with the water bag full and up to 14 min with the bag empty (see Figure 3). However, the hardware solution described here is a prototype and a larger UAV platform could allow for greater payloads and longer flight times more suitable for real missions.

---

[5] https://ctu-mrs.github.io/docs/hardware/motor_tests.html
[6] https://terabee.b-cdn.net/wp-content/uploads/2020/05/evo-thermal_specsheet.pdf - Accessed: 03-30-2022
[7] http://static.garmin.com/pumac/LIDAR_Lite_v3_Operation_Manual_and_Technical_Specifications.pdf
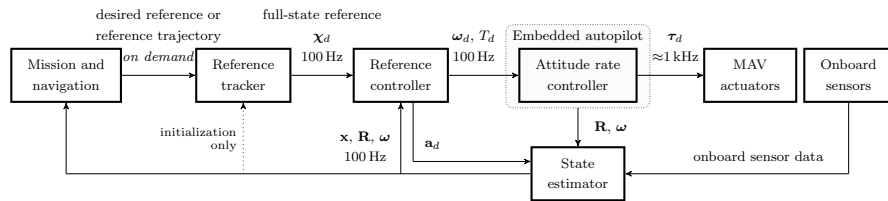- Accessed: 03-30-2022

414 · Walter et al.

Figure 4. A diagram of the control system architecture. A *Mission and navigation* software supplies a 3D position and heading reference $(\mathbf{r}_d, \eta_d)$ or a time-parametrized reference trajectory $\{(\mathbf{r}_d, \eta_d)_1, (\mathbf{r}_d, \eta_d)_2, \ldots, (\mathbf{r}_d, \eta_d)_k\}$ to a reference tracker. The *Reference tracker* creates a smooth and feasible reference χ_d for a reference feedback controller. The feedback *Reference controller* produces desired thrust and angular velocities (T_d, ω_d) for the Pixhawk embedded flight controller. A *State estimator* fuses data from onboard sensors to create an estimate of the UAV translation and rotation $(\mathbf{x}, \mathbf{R}, \omega)$.

5. Software System Structure

This section describes the software components of the proposed system. All of the detailed components are executed on the onboard *Intel NUC-i7* PC.

5.1. Control and estimation of the UAV state

The UAV is controlled by a multi-layer control pipeline as depicted in Figure 4. Since the focus of this paper is for outdoor firefighting, only the basic structure of the control architecture will be described here. For a more detailed description of the whole control software platform, we refer the reader to (Baca et al., 2021). The desired trajectory supplied by a *Mission and Navigation* module (in our case, by a high-level motion planning and fire extinguishing procedure) is first processed by the *Reference tracker* (Baca et al., 2018) based on model predictive control to obtain a smooth and feasible reference for the *Reference controller*. The *Reference controller* uses this reference to provide the SE(3) geometric state feedback control (Lee et al., 2010) of the translational dynamics and orientation of the UAV. The attitude rate and thrust commands generated by the *Reference controller* are sent to the embedded *Attitude rate controller* in the flight control unit of the UAV. The feedback loop of the *Reference controller* is closed by the *State estimator*, which fuses data from onboard sensors with the UAV attitude to obtain a precise and reliable state estimate.

The state estimation process uses Kalman filtering to estimate the 3-D position of the UAV and its heading angle, along with their respective first and second derivatives. The UAV state is divided into lateral, altitude, and heading parts. Such decoupling facilitates tuning of the filter and smaller system matrices save computation resources. The lateral filter uses position corrections from GNSS and heading filter corrections from magnetometers. The altitude estimation fuses data from the built-in barometer with measurements from the laser rangefinder.

5.2. Motion planning and exploration

The precise positions of the fire objects are unknown ahead of the mission, therefore the facade must be surveyed for their localization. The strategy used here was to allow each UAVs to circumnavigate the building at altitudes corresponding to individual floors at a distance of 4.5 m while aiming their thermal cameras towards the walls of the building. This gives us thermal coverage (see section 6.1 for the thermal vision) of a vertical strip of the building facade of 9.5×2.7 m due to the extended vertical FoV of the three thermal cameras. The detection from the RGB camera (see section 6.2) is also active, but in this phase it is more likely that the fire objects are located using thermal vision due to the greater robustness of thermal vision in this subtask.

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES
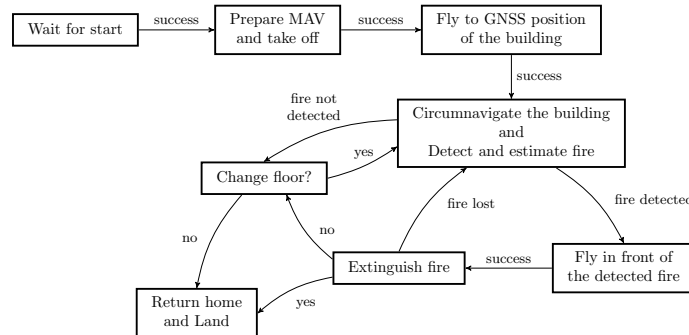
Figure 5. A high-level diagram of the main state machine. This figure shows the behavior and decision-making implemented on the UAV for the facade firefighting from takeoff to return.

Once the fire object is located, the UAV stops circumnavigating and flies in front of the object at the distance of 1.5 m from its estimated center along the normal of the corresponding wall. After reaching this position, control is handed over to the extinguishing sub-system (see section 6.4). As long as the fire object is not lost, the UAV depletes all the water being carried during the extinguishing maneuver. If the target is lost, the UAV continues to search for fire objects. After depletion of the water, the UAV flies back to its starting position and lands. In the case of circumnavigating the building without detecting a fire object, the UAV can change the flying altitude and start the search for the fires again or it can return to its starting position to land (depending on user defined settings). The latter option was not used during the competition. The complete strategy of the task is shown in the diagram in Figure 5.

In the competition, multiple UAVs are deployed in the same environment and may possibly collide with each other. To avoid this, the UAVs share their future trajectories to detect possible collisions and react to them. In the case of a threat of collision, the UAVs have different set priorities and the one with lower priority will replan its trajectory by ascending to a higher flying altitude. This collision avoidance technique is explained in detail in (Baca et al., 2018).

6. Detection, localization, and extinguishing of fire objects

The most easily detected feature of the fire objects are the real flames produced from their gas pipe ring. These are shown in the image of the thermal cameras as large blobs of more than 200 °C and if directly observed, they can be consistently detected from a significant distance. Depending on various factors such as air currents and water being deployed, their appearance in the thermal images can morph significantly (see Figure 6). The flames are observed by the thermal cameras as opaque and significantly hotter than the gas pipe rings that produce them. They are in fact so much hotter than the central aluminium heating element, that the element itself is practically undetectable (although it does occasionally visibly reflect the heat of the flames). The elements are heated to 120 °C, but our cameras observe them as having approximately 70 °C, due to the material having the emissivity value of 0.55 (Minkina and Dudzik, 2009), as opposed to the value of 0.95 that our cameras[8] presume.

Given how the opening that collects the deployed water for evaluation is smaller than the typical area of the observed flames and is obscured by them, it is nearly impossible to aim accurately into the opening based on the thermal images alone (see Figure 9 - A1, A2 for example of unsuitable thermal

---

[8] https://terabee.b-cdn.net/wp-content/uploads/2020/05/evo-thermal_specsheet.pdf - Accessed: 03-30-2022

3.2. ARTICLE 5: AERIAL LIQUID-BASED REAL FIRE EXTINGUISHING
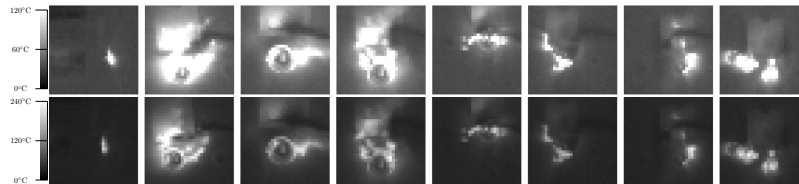
416 · Walter et al.

Figure 6. Thermal images of the fire objects taken with our thermal camera shown in two different thermal ranges. With the exception of the first image, these were taken within 30 s and the flames themselves are always visible. Note how different the fire object appears due to the effects of random air flow, pressure from the presence of a UAV, the water stream (visible in some images as a darker band starting from the right where the nozzle was), and other factors.
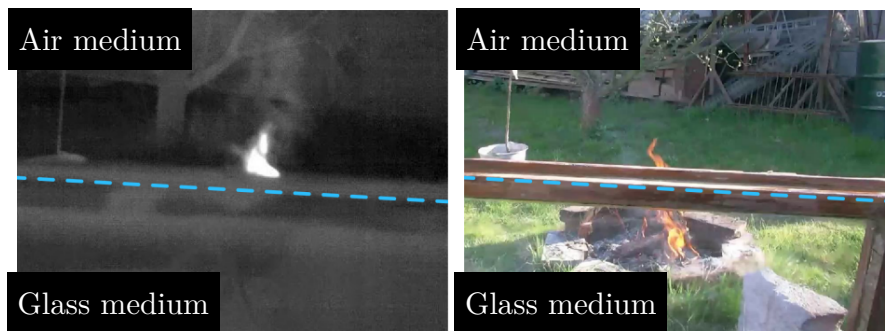
Figure 7. An example of a significant limitation of thermal vision that RGB vision can address: thermal cameras do not see heat sources through glass, posing a hurdle when searching for fires behind windows. This motivates the use of fusion of RGB and thermal vision in robotic firefighting even for the purpose of real deployment. While such conditions were beyond the scope of the competition where the target was only a small, specific region behind flames, the RGB vision can be adapted for detection of fires behind surfaces opaque to infrared radiation as a fallback.

image), despite it being possible to temporarily extinguish the flames. While in theory an applicable strategy could be to first extinguish the flames and then to aim at the aluminium heating element in the window when flames are not present, this would waste a significant portion of the limited amount of carried water and time and would additionally raise the requirement for dynamically changing detection parameters. Instead, we have decided to combine two modes of vision - thermal and RGB - to address their relative drawbacks and benefit from the strengths of both. We have based the additional vision system on an onboard RGB camera to enable aiming at the center of the opening for collecting water when the UAV has sufficiently closed in on the fire object. Using such hybrid vision in firefighting is not merely a reaction to the conditions of the competition, it is additionally motivated by the real-world problem that thermal cameras can not see through glass that blocks infrared (IR) radiation (see Figure 7) while RGB cameras can compensate by detecting fires through windows.

6.1. Thermal vision

The primary goal in the design of the thermal vision subsystem was to localize the center of the opening for water collection in the images from thermal cameras. We have observed in our tests

that the flame seen in the thermal image often covers the center we are attempting to target and that the central aluminium heating element is often too dim by itself to be consistently detected. This was made more challenging by the fact that the small resolution of the lightweight thermal cameras does not lend itself well to advanced computer vision algorithms. These observations led us to develop a simple technique for localizing a "best guess" image position for targeting. This technique was designed after surveying previously recorded datasets of thermal images of the real fire objects. From these images, we selected samples from such frames where simultaneously:

- The flame was not split into multiple individual flames fully separated by colder regions
- The colder interior of the gas pipe ring was fully enclosed by the higher temperature of the piping and flames,
- The aluminium heating element is visible and not connected to the surrounding flames and piping.

For these frames, we tuned a double temperature threshold, such that the binarized image $\mathbf{B_{temp}}$ of pixels from the thermal image $\mathbf{I_I}$ within these two values makes the pipe and flames a distinct contour from the aluminium heating element. In order to emphasize small objects such as the aluminium heating element, we additionally AND-combine the binarized image with thresholded matrix $\mathbf{B_{diff}}$ of the thermal image Laplacian $\mathbf{I_L}$ (representing a required local differential). In our case, these thresholds were as follows:

- Temperature must be above 60 °C - this selected only hot objects in the image
- Temperature must be below 250 °C - this is meant to generate additional negative contours in the hottest areas of the largest flames
- The local differential must be above 10 °C - this emphasized thin objects, such as the gas pipe ring and the heating element to increases robustness to ambient temperature shifts.

We then selected the centroid $[x, y]$ of the innermost contour $\mathcal{D}_{sel}$ as the image target position, regardless of whether it is a positive or negative value in the binarized image, as long as the contour is not the background itself. By the innermost contour, we mean to say the one which is enclosed within the most other contours. If multiple contours $\mathcal{D}_L$ enclosed by this number of other contours exist, we select the one with the largest area. This thermal image processing approach is summarized in the Algorithm 1. The algorithm leads to one of three cases of target selection, as follows: In the rare case that the image satisfies the requirements we had for the sample images, the output image position will correspond to the heating element in the center of the opening where we would ideally deploy water (Figure 8 - A1,A2). Otherwise, when the interior of the pipe ring is fully enclosed in an area of higher temperature, but the aluminium heating element is either not visible (Figure 8 - B1) or seen as connected to the surrounding higher temperature, the innermost contour will be the centroid of the enclosed darker section. Although it is not in the center, this is still inside the opening (Figure 8 - B2) which would usually lead to the successful deployment of water. Lastly, when the colder center is not visible or not fully enclosed by a hotter temperature, the output will be the centroid of the hotter visible area (Figure 8 - C1,C2) and will lead to water being deployed into the flames themselves to temporarily extinguish them. This in turn occasionally leads to the observed situation being closer to the sample images, due to the flames restarting slowly. As this occurs, the ring tends to be more clearly visible as enclosing the colder central region. This third case is also the closest to real firefighting, where the goal is not to precisely target a small, difficult to see area inside of a flame, but rather to actually extinguish the easier to detect flame itself. Aiming for the centroid of the heat silhouette is a good strategy for extinguishing the flame by cooling down its core. Additionally, from a distance the fire object almost always appears as a simple continuous contour (Figure 8 - C1). This detection represents the direction towards the flame and is useful for the phases of flight where the UAV is searching for new fire objects.

Algorithm 1. Retrieval of the estimated center of fire object from thermal images

1: **function** GET_THERMAL_TARGET_IN_IMAGE
2:     $\mathbf{I_I} \leftarrow$ GET_LATEST_THERMAL_IMAGE   ▷ 32×32 matrix of Celsius temperatures
3:     $\mathbf{K_L} \leftarrow \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$ ▷ Convolution kernel for approximating spatial second derivative
4:     $\mathbf{I_L} \leftarrow$ CONVOLVE($\mathbf{I_I}, \mathbf{K_L}$)   ▷ 32×32 Laplacian image matrix
5:     $\mathbf{I_L} \leftarrow \mathbf{I_L} \cdot 0.25$ ▷ Per-element scaling to account for the integer values of $\mathbf{K_L}$
6:     $\mathbf{B_{temp}} \leftarrow (250.0°C > \mathbf{I_I} > 60.0°C)$ ▷ Logical binary 32×32 matrix -
    $\forall i, j : B_{temp_{ij}} = (250.0°C > I_{I_{ij}} > 60.0°C)$
7:     $\mathbf{B_{diff}} \leftarrow \mathbf{I_L} > 10.0°C$ ▷ Logical binary 32×32 matrix - $\forall i, j : B_{diff_{ij}} = (I_{L_{ij}} > 10.0°C)$
8:     $\mathbf{B_{comb}} \leftarrow \mathbf{B_{temp}} \wedge \mathbf{B_{diff}}$ ▷ Per matrix element logical AND
9:     $\mathcal{C} \leftarrow$ GET_CONTOURS($\mathbf{B_{comb}}$)   ▷ Selects elements $B_{comb_{ij}}$ with neighbors of
    different value and groups their indices into sets of 8-adjacent *true* elements and 4-adjacent *false* elements:
    $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, ...\}, \mathcal{C}_N = \{[i_{N1}, j_{N1}], [i_{N2}, j_{N2}], ...\}$
10:     $\mathbf{h} \leftarrow 0_{|\mathcal{C}|}$ ▷ Initialize hierarchy vector for count of enclosing contours per contour: $\mathbf{h} = [h_1, h_2, ..., h_N]$
11:     $m_h \leftarrow 0$ ▷ An integer to track the current maximum enclosing count
12:     **for all** $\mathcal{C}_N \in \mathcal{C}$ **do**
13:         **for all** $\mathcal{C}_M \in \mathcal{C}$ **do**
14:             **if** $M \neq N$ **then**
15:                 **if** IS_ENCLOSING($\mathcal{C}_M, \mathcal{C}_N$) **then**   ▷ Check if $\mathcal{C}_M$ totally encloses $\mathcal{C}_N$
16:                     $h_N \leftarrow h_N + 1$ ▷ Increment the appropriate element of hierarchy vector
17:                     **if** $h_N > m_h$ **then**
18:                         $m_h \leftarrow m_h + 1$
19:                     **end if**
20:                 **end if**
21:             **end if**
22:         **end for**
23:     **end for**
24:     $\mathcal{D} \leftarrow \{\mathcal{C}_K \in \mathcal{C} : |\mathcal{C}_K| = m_h\}$ ▷ Get the innermost contours
25:     $\mathcal{D}_{sel} \leftarrow \underset{\mathcal{D}_L \in \mathcal{D}}{\text{argmax}} |\mathcal{D}_L|$ ▷ From the innermost contours, select the largest
26:     $[x, y] \leftarrow \frac{\sum \mathcal{D}_{sel}}{|\mathcal{D}_{sel}|}$ ▷ Approximate contour centroid as the average x and y coordinate of its elements
    return $[x, y]$
27: **end function**   ▷ Note: $|\mathcal{D}|$ denotes the number of elements in set $\mathcal{D}$
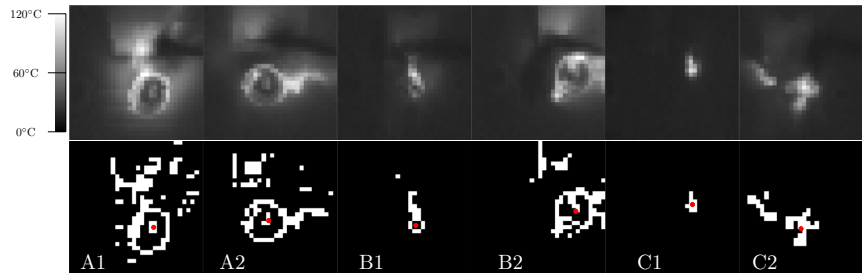
Figure 8. Thermal views (top row) and contour processing (bottom row) of the fire object for the three mentioned cases. The red dot denotes the selected contour centroid on output. A1 and A2 show cases where both the colder enclosed center of the ring and the aluminium heating element are visible. B1 and B2 are cases when only the colder center can be retrieved. C1 and C2 show the most common case where we may only directly see continuous individual flames and thus select the largest of them.

CHAPTER 3.   IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND
STABILIZATION TECHNIQUES

6.2. RGB vision

Since the goal of the competition was to deposit the maximum amount of water into the openings of the fire objects as opposed to extinguishing the flames, the fire itself poses a challenge in this task. Its shape and size is constantly changing under the influence of various factors such as wind, air currents from the UAV, local fuel distribution, and the extinguishing procedure. Additionally, as mentioned above, it obscures static features that we are specifically interested in localizing. Specifically, in order to locate the center of the opening for collecting water, we had to detect and localize the opening itself, the aluminium heating element, or the gas pipe ring that surrounded it. The challenge for RGB vision was, therefore, primarily in how to consistently detect the static features despite the presence of flames. We had decided to focus our detection on the gas pipe rings. They were the largest feature to surround the opening and possessed a shape that would allow us to easily retrieve the center of the opening itself. In order to see "through" the flame when it obscured the gas pipe ring and center of the opening, we exploited our observation that the flames were mostly transparent to blue light. Therefore, we have used only the blue channel image matrix $\mathbf{I_B}$ to detect the contour of the gas pipe ring around the water collection opening.

The information from the other color channels was used for retrieval of the contours of the flames (see Figure 9 - C for extracted flame contours). Even though they were not sufficient for precise aiming, their detection was trivial and allowed the system to crop out the axis-aligned rectangle
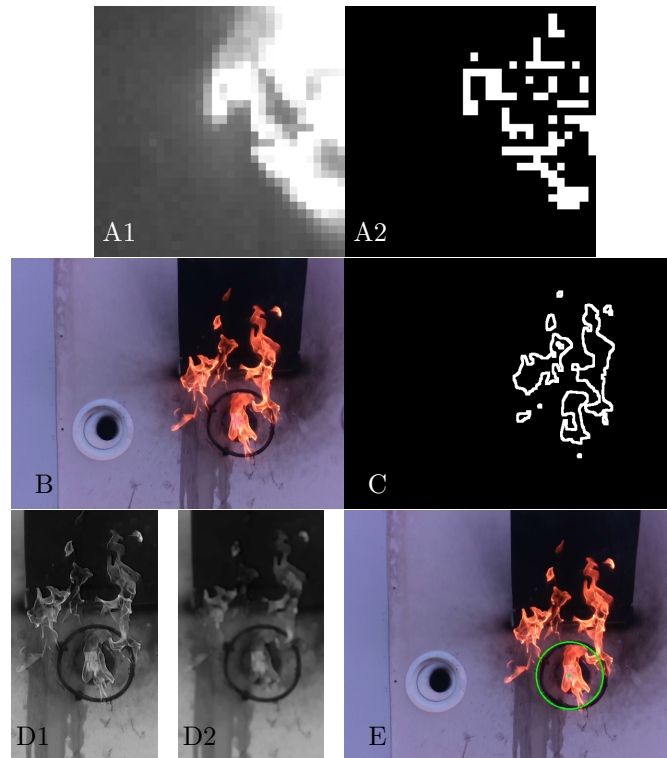

Figure 9. Steps of target detection with RGB vision in the case that the thermal vision is unsuitable for aiming.

420 · Walter et al.

$[s_x, s_y, w_x, w_y]$ within, which was where the gas pipe ring and the opening were most likely located (see Figure 9 - D1). This rectangle was not a tight bounding box only enclosing the flames, but rather a rectangle which contained the flames and took into account the possible influence of external disturbances. The bounding box was used to select a sub-matrix $\mathbf{I_{Bsub}}$ from the blue channel image $\mathbf{I_B}$ for further processing, thus saving processing power and reducing the chance of false detections. The image $\mathbf{I_{Bsub}}$ was blurred slightly (see Figure 9 - D2) to suppress traces of the flames still visible in the form of high-image frequency features. The Canny Edge Detector was used to subsequently produce a binary image $\mathbf{B_{Bsub}}$. Next, contour detection was applied on the image $\mathbf{B_{Bsub}}$. For each positive contour, $\mathcal{D}_M$ its smallest enclosing circle with a center at $[s_{cx}, s_{cy}]$ and radius s_{cr} was calculated. Comparison of these circles with their corresponding contours has been done using area comparison and the IOU (intersection-over-union) approach. Additional filtering was applied, taking into account the approximate expected image area of the circle t_a based on the physical dimensions of the gas pipe ring and the distance to the wall d that was retrieved from LIDAR readings. If the contour $\mathcal{D}_M$ passes through the filtering described above, the center of the enclosing circle $[s_{cx}, s_{cy}]$ is used as the image position of the extinguishing target. For cases when more than one compliant contour existed in the image (meaning some object in addition to the opening was still in consideration), an additional strategy had to be applied. The following were considered:

- Accept the circle with the center closer to the center of the flame bounding box. This may be challenging when the wind level is high and, therefore, the flames are moving further away from the opening
- Discard the detection with a number of enclosing circles above one. Depending on lightning conditions, this may lead to a significant period without detection of the opening, even when the UAV is in front of it
- Track the previous detections of the opening and select the enclosing circle with a center that is closer to previous detections. This case presumes that the UAV is not moving with respect to the fire object. Alternatively, such motion must be taken into account and may be challenging due to multiple factors, including wind influence, GNSS inaccuracy, or computational complexity. Such factors make it impossible to include this data within the real-time factor. Additionally, if a number of previous readings were wrong, this may lead to further error accumulation.

The third approach was chosen for the competition. When the number of enclosing circles was equal to one, all previous readings were discarded allowing us to overcome the issue of error accumulation. The steps of the visual detection approach outlined here are expanded upon in the Algorithm 2. After the enclosing circle was selected, its center was taken as the center of the opening for water collection (see Figure 9 - E) and the fire extinguishing was directed accordingly.

6.3. Fire localization in 3D

We assumed that the thermal cameras conform to the pinhole camera model and derived their focal distance from the pixel resolution w per side and their FoV ϵ per side. Upon detection of the lowest level contour in the thermal image, we calculate the average image coordinates of the pixels of the contour as x and y. These coordinates are converted to direction vectors using the assumed camera model as

$$\mathbf{v_t} = \begin{bmatrix} v_{t_x} \\ v_{t_y} \\ v_{t_z} \end{bmatrix} = \begin{bmatrix} 1/f & 0 & -((w-1)/2)/f \\ 0 & 1/f & -((w-1)/2)/f \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \text{ where } f = \frac{(w/2)}{\tan(\epsilon/2)}. \tag{1}$$

Subsequently, we normalize $\mathbf{v_t}$ and transform it into a coordinate frame centered in the optical center of the camera, with the x-axis pointing forwards, the y-axis to the left, and the z-axis upwards. We call this coordinate frame the thermal base frame. The direction vector after this transformation is denoted as $\hat{\mathbf{v_f}}$.

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

Extinguishing real fires by fully autonomous multirotor UAVs in the *MBZIRC 2020* competition · **421**

Algorithm 2. Retrieval of the estimated center of a fire object from RGB images

1: **function** GET_RGB_TARGET_IN_IMAGE
2:   $\mathcal{O} \leftarrow \emptyset$ ▷ initialize output set
3:   $\mathbf{I_I} \leftarrow$ GET_LATEST_RGB_IMAGE   ▷ 3-channel color image matrix from camera
4:   $[\mathbf{I_R}, \mathbf{I_G}, \mathbf{I_B}] \leftarrow$ SPLIT_RGB_CHANNELS($\mathbf{I_I}$)   ▷ $\forall i, j, X \in R, G, B : I_{Xij} \in \{0, ..., 255\}$
5:   $[\mathbf{I_H}, \mathbf{I_S}, \mathbf{I_V}] \leftarrow$ CONVERT_TO_HSV($\mathbf{I_I}$) ▷ Express the input in terms of hue, saturation and value images: $\forall i, j : I_{Hij} \in (0, 2\pi)\,;\, I_{Sij} \in (0, 1)\,;\, I_{Vij} \in (0, 1)$
6:   $\mathbf{B_{flame}} \leftarrow (0 < \mathbf{I_H} < 1.22) \wedge (0.52 < \mathbf{I_S} < 1) \wedge (0.40 < \mathbf{I_V} < 1)$ ▷ Binary matrix with *true* at elements where the HSV values fall within an empirically obtained ranges for flames
7:   $\mathbf{B_{flame}} \leftarrow$ MORPHOLOGY_OPEN($\mathbf{B_{flame}}, \mathbf{J}_3$) ▷ Morphological opening with structuring element of 3×3 matrix of ones to suppress small random blobs
8:   $\mathcal{C} \leftarrow$ GET_POSITIVE_CONTOURS($\mathbf{B_{flame}}$) ▷ Selects elements where $B_{flame_{ij}} = true$ with neighbors of *false* value and groups their indices into 8-adjacent sets: $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, ...\}, \mathcal{C}_N = \{[i_{N1}, j_{N1}], [i_{N2}, j_{N2}], ...\}$
9:   $\mathcal{C}_{sel} \leftarrow \underset{\mathcal{C}_M \in \mathcal{C}}{\mathrm{argmax}}\, |\mathcal{C}_M|$ ▷ Select the largest contour
10:   $[s_x, s_y, w_x, w_y] \leftarrow$ GET_AABB($\mathcal{C}_{sel}$) ▷ Smallest axis-aligned bounding box of $\mathcal{C}_{sel}$ with center $[s_x, s_y]$ and dimensions $[w_x, w_y]$
11:   $w_x \leftarrow 1.2 \cdot w_x$ ▷ Expand the horizontal range to account for flame scatter by wind
12:   $w_y \leftarrow$ GET_VERTICAL_DIMENSION($\mathbf{I_I}$)   ▷ Expand the vertical range to account for flame buoyancy
13:   $s_y \leftarrow 0.5 \cdot$ GET_VERTICAL_DIMENSION($\mathbf{I_I}$)   ▷ Shift center to cover the entire image height
14:   $\mathbf{I_{Bsub}} \leftarrow$ SUBMATRIX($\mathbf{I_B}, [s_x, s_y, w_x, w_y]$)   ▷ Section of $\mathbf{I_B}$ within the bounding box
15:   $\mathbf{I_{Bsub}} \leftarrow$ CONVOLVE($\mathbf{I_{Bsub}}$, GAUSSN_KERNEL(5)) ▷ Apply Gaussian blur with kernel size of 5
16:   $\mathbf{B_{Bsub}} \leftarrow$ CANNY($\mathbf{I_{Bsub}}$, 35, 93)   ▷ Apply the Canny Edge Detector with the empirically discovered hysteresis thresholds
17:   $\mathbf{B_{Bsub}} \leftarrow$ MORPHOLOGY_DILATE($\mathbf{B_{Bsub}}, \mathbf{J}_3$) ▷ Morphological dilation with structuring element of 3×3 matrix of ones
18:   $\mathcal{D} \leftarrow$ GET_POSITIVE_CONTOURS($\mathbf{B_{Bsub}}$)   ▷ $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, ...\}, \mathcal{D}_M = \{[i_{M1}, j_{M1}], [i_{M2}, j_{M2}], ...\}$
19:   **for all** $\mathcal{D}_M \in \mathcal{D}$ **do**
20:     $[s_{cx}, s_{cy}, s_{cr}] \leftarrow$ MIN_ENCLOSING_CIRCLE($\mathcal{D}_M$) ▷ Center and radius of smallest enclosing circle
21:     $a_M \leftarrow$ CONTOUR_INNER_AREA($\mathcal{D}_M$)   ▷ Number of elements enclosed by $\mathcal{D}_M$
22:     $a_C \leftarrow \pi \cdot {s_{cr}}^2$ ▷ Area of the enclosing circle
23:     $f \leftarrow$ GET_FOCAL_LENGTH   ▷ Camera focal length in pixels
24:     $d \leftarrow$ GET_LIDAR_WALL_DISTANCE   ▷ Distance from the wall based on onboard LIDAR
25:     $t_a \leftarrow \pi \cdot \left(f \cdot (0.125^\circ C / d)\right)^2$ ▷ Expected image area inside the gas pipe ring based on its radius and observing distance
26:     $v_{IoU} \leftarrow$ IoU(GET_AABB($\mathcal{D}_M$), $[s_{cx}, s_{cy}, 2 \cdot r, 2 \cdot r]$)   ▷ Intersection over union of rectangles
27:     $v_{adiff} \leftarrow \frac{|a_M - a_C|}{a_C}$ ▷ Relative area difference
28:     **if** $(a_M > 0.7 \cdot t_a) \wedge (v_{IoU} > 0.8) \wedge (v_{adiff} < 0.3)$ **then**
29:       $\mathcal{O} \leftarrow \mathcal{O} \cup \{s_{cx}, s_{cy}\}$ ▷ Add the current enclosing circle center
30:     **end if**
31:   **end for**
32:   **if** $|\mathcal{O}| > 1$ **then**
33:     $\mathcal{O}_{prev} \leftarrow$ GET_PREVIOUS_DETECTIONS
34:     $\mathcal{O} \leftarrow \{$GET_CLOSEST_DETECTION($\mathcal{O}, \mathcal{O}_{prev}$)$\}$ ▷ Member of $\mathcal{O}$ closest to any member of $\mathcal{O}_{prev}$
35:   **else if** $|\mathcal{O}| = 1$ **then**
36:     DISCARD_PREVIOUS_DETECTIONS   ▷ To prevent long-term accumulation of error
37:   **end if**
38:   STORE_AMONG_PREVIOUS_DETECTIONS($\mathcal{O}$)
39: **return** $\mathcal{O}$
40: **end function**   ▷ Note: $|\mathcal{D}|$ denotes the number of elements in set $\mathcal{D}$

For the extinguishing action done by UAVs, we also needed a distance estimate. This was achieved by combining the direction vectors with a source of surface shape measurement. We used the onboard 2D LIDAR to estimate the outline of the objects in front of the cameras.

It should be noted that the LIDAR output data and the direction vector $\hat{\mathbf{v}}_\mathbf{f}$ have to be expressed in a common coordinate fame. This was trivial due to our knowledge of the rigid UAV geometry, and we have chosen to transform all of these into the frame of the front-facing thermal camera. The
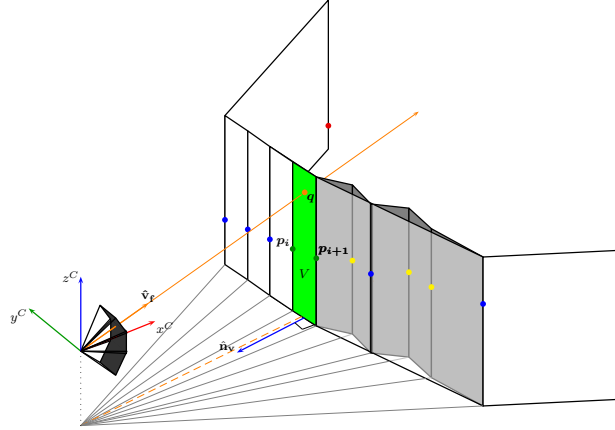
422 · Walter et al.

Figure 10. The 3D localization of fires on a wall based on vision and 2D LIDAR data.

time-delays between the different measurements were negligible compared to the UAV dynamics, with the output rate of the LIDAR being 20 Hz.

The output of the sensor was a set of coplanar 3D points $\mathcal{P}_{lidar}$ ordered by their angle from the sensors (blue, green, and yellow points in Figure 10). Given that the targets were attached to a flat wall, we can presume that the outline points retrieved by the LIDAR represent the shape of the surface sufficiently well. However, the set of outline points must first be reduced to their convex hull in terms of their horizontal coordinates in order to remove measurements behind windows if they were captured by the LIDAR. This is necessary, since the fire objects can be located above or below windows, in which case if we tried to obtain their 3D positions, then they may be incorrectly evaluated as being on an interior wall of the building. The convex hull is obtained by a pseudo gift-wrapping algorithm as follows: we first remove from $\mathcal{P}_{lidar}$ the points that are outside of the horizontal field of view of the thermal cameras. Next, we expand $\mathcal{P}_{lidar}$ with two new points beyond the leftmost and rightmost of the remaining points in terms of their angle from the LIDAR. These points (red points in Figure 10) are generated by copying the previous leftmost and rightmost points and increasing their distance while preserving their bearing. We then select the three leftmost points (the first of which was just newly added) and check that they form a left-turning angle. If that is the case, we select the next triplet by sliding one point to the right and repeating the check on the next triplet. If a right turning point is found, the middle of the three points is discarded from $\mathcal{P}_{lidar}$. This process is repeated until no right turns remain and we are left with a convex hull where interior points (yellow points in Figure 10) were removed. We can now describe the surface in front of the cameras as an extrusion object composed of vertical plane segments defined by pairs of adjacent points remaining in $\mathcal{P}_{lidar}$ (white and green surfaces in Figure 10). For each of the direction vectors retrieved from the thermal cameras, we select the vertical plane V (green plane in Figure 10) corresponding to the generating pair of 3D points $\boldsymbol{p}_i$ and $\boldsymbol{p}_{i+1}$ (green points in Figure 10) from $\mathcal{P}_{lidar}$ that it passes in between of in terms of the yaw angle. We select this pair of points as

$$\{\boldsymbol{p}_i \in \mathcal{P}_{lidar} \mid \text{atan2}(p_{i_y}, p_{i_x}) > \text{atan2}(v_{f_y}, v_{f_x}) \wedge \text{atan2}(p_{i+1_y}, p_{i+1_x}) < \text{atan2}(v_{f_y}, v_{f_x})\}. \quad (2)$$

The 3D position of the estimated target is then obtained by calculating intersection point $\boldsymbol{q}$ of the line (the orange line in Figure 10) defined by $\hat{\mathbf{v}}_\mathbf{f}$ passing through the origin with the selected vertical plane V. The plane V is defined by a point and a normal vector, specifically in this case the

point $\boldsymbol{p}_i$ and the normal vector $\mathbf{n_v}$, obtained as

$$\mathbf{n_v} = \begin{bmatrix} -(p_{i+1_y} - p_{i_y}) \\ p_{i+1_x} - p_{i_x} \\ 0 \end{bmatrix}. \tag{3}$$

The intersection $\boldsymbol{q}$ is then calculated as

$$\boldsymbol{q} = \hat{\mathbf{v}}_{\mathbf{f}} \cdot t, \tag{4}$$

where t is obtained using normalized vector of the surface normal $\hat{\mathbf{n}}_{\mathbf{v}}$ as

$$t = \frac{\hat{\mathbf{n}}_{\mathbf{v}} \cdot \boldsymbol{p}_i}{\hat{\mathbf{n}}_{\mathbf{v}} \cdot \hat{\mathbf{v}}_{\mathbf{f}}}. \tag{5}$$

The estimate of the surface normal is further used to steer the UAV to the ideal perpendicular position for extinguishing located $r_d = 1.5$ m from the fire object position estimate along the normal.

We do not consider single detection of the 3D position to be sufficient. Instead, we implemented a Kalman filter that stores multiple measurements as an array of trackers and refines each tracker state using new measurements.

The state vector of the Kalman filter used here is

$$\boldsymbol{q_k} = \left[q_{k_x}, q_{k_y}, q_{k_z}, q_{k_\eta}\right]^T, \tag{6}$$

where q_{k_x}, q_{k_y}, and q_{k_z} are the coordinates of the fire object in the world coordinate frame. The q_{k_η} represents the azimuth of the surface normal for that fire object. The filtering mechanism stores multiple such state vectors that are corresponding to multiple different detected fire objects. We update a specific state vector $\boldsymbol{q_k}$ using a new estimate of the fire object position $\boldsymbol{q}$ and normal $\mathbf{n_v}$, if $\boldsymbol{q}$ is closer in the world frame than 1 m to $\left[q_{k_x}, q_{k_y}, q_{k_z}\right]^T$ and at the same time the horizontal component of $\mathbf{n_v}$ is closer than $90°$ to q_{k_η}.

In that case, the state is refined in the standard Kalman filter correction step. Otherwise, we generate a new tracker from the measurement. To account for random errors, a state is only considered validated if at least 10 measurements have been associated with it. Additionally, the states have a decay time to discard old estimates after 10 s without updates.

Note, that the 3D pose estimation of the target position does not depend on the global self-localization of the UAV itself, as it directly uses the measurements from onboard sensors. However, we do use global localization of the UAV in order to convert the measured intersection $\boldsymbol{q}$ and its normal $\mathbf{n_v}$ into the world coordinate frame after calculating them. The main goal of this transformation is for the Kalman filter to better cope with the influence of the ego-motion of the observing UAV on the filter state vector $\boldsymbol{q_k}$. The global localization of the UAV is based on a sensor fusion from GNSS and auxiliary sensors, such as an altimeter and the inertial sensor built into the flight controller (Baca et al., 2021).

The Kalman filter brings three main benefits:

- The discarding of random errors
- The refinement of estimates as we approach the targets
- Preserving targets in memory (even if they are currently out of view).

The last point is especially significant as our thermal cameras only cover $33°$ horizontally at any moment. In this way, we can use their extended total vertical coverage angle to sweep the environment for fire objects.

6.4. Fire extinguishing

Upon obtaining the first validated fire object detection state in the Kalman filter array, the UAV is sent to a position 1.5 m in front of the given target along the estimated normal. The observation

424 · Walter et al.

angles with respect to the front wall from which the fire objects can be reliably seen are limited and the observation, as well as the extinguishing, has to be done from as close to a perpendicular position to the wall as possible. It is further useful to maintain perpendicular alignment for extinguishing as this maximizes robustness of the correct aiming with respect to drifting in an arbitrary direction.

As the UAV flies to the designated position, its estimate of the target position and surface normal improve thanks to obtaining new detections. Once it reaches the position, the control is given to the fire extinguishing sub-system. In this state (see Algorithm 3 for a pseudocode on the UAV behavior), the UAV is steered towards a specific position with respect to the detected fire object, defined the same way as the position to which the UAV was sent previously at the desired distance r_d of 1.5 m from the target along the estimated surface normal. However, continuously driving the UAV to this exact position, especially in the face of potential fire object estimation errors and disturbances such as air currents would lead to rapid tilting in attempts to correct the current position, especially close to the desired position. In our intial experiments, this lead to the direction of the water stream being significantly unstable, since the water nozzle is rigidly attached to the body and thus follows the tilts. To address this issue, we have included a hysteresis to the steering. Specifically, we have defined two ranges of angle and distance offsets. These are illustrated in Figure 12. The angle ranges were defined as limits to angles formed by the surface normal and the line connecting the center of the fire object and the UAV - the inner range α_i was set to $\pm 5°$ and the outer range α_o to $\pm 10°$. The distance ranges are offsets from the desired extinguishing distance of 1.5 m, and these were set to ± 0.075 m for the inner range r_i and ± 0.15 m for the outer range r_o. Once the UAV has reached the inner ranges (red zone in Figure 12, it is forbidden to correct its horizontal (X-Y) coordinates despite any disturbances and it can only correct its altitude and heading as changing these does not generate tilt in the UAV. The UAV thus tends to drift or "float". Continuously correcting the
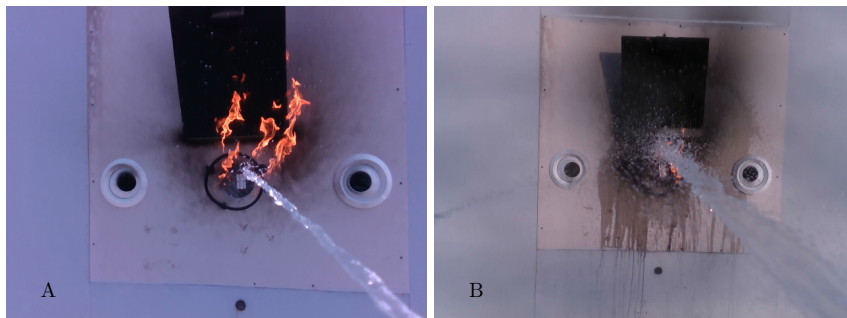

Figure 11. An onboard view of the fire extinguishing in the first run of the competition (A) and in the final exhibition (B).
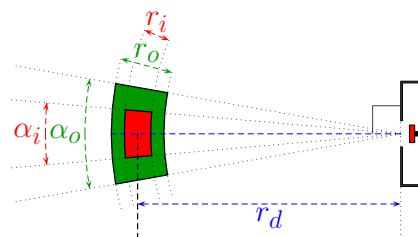

Figure 12. Steering hysteresis for fire extinguishing viewed from the top.

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

Algorithm 3. High-level control during fire extinguishing

1: **function** EXTINGUISH_FIRE
2:      $t_{qk} \leftarrow$ GET_CURRENT_TIME   ▷ Initialize the pose estimate time stamp
3:      $W \leftarrow false; H \leftarrow false$ ▷ Initialize water pump state to off and the pose hysteresis to Controlling
4:      $w_t = w_{ti}$ ▷ Initialize the remaining water spraying time to measured value (25 s)
5:      **while** $w_t > 0$ & IS_RECENT(t_{qk}) **do**   ▷ Until the water is depleted or the target tracking is lost
6:          $q_k, t_{qk} \leftarrow$ GET_LATEST_FIRE_ESTIMATE   ▷ $\left[q_{k_x}, q_{k_y}, q_{k_z}, q_{k_\eta}\right]^T$, time stamp
7:          $v_t, t_{vt} \leftarrow$ GET_LATEST_THERMAL_DIRECTION   ▷ Target direction form thermal vision, time stamp
8:          $v_v, t_{vv} \leftarrow$ GET_LATEST_RGB_DIRECTION   ▷ Target direction form RGB vision, time stamp
9:          $o_p \leftarrow$ GET_LATEST_ODOMETRY   ▷ UAV pose in world frame $\left[o_{p_x}, o_{p_y}, o_{p_z}, o_{p_\eta}\right]^T$
10:         $W_p \leftarrow W$ ▷ Store previous water pump state
11:         $\hat{\mathbf{n}}_{\mathbf{vk}} \leftarrow$ ROTATE_ABOUT_Z$([1, 0, 0]^T, q_{k_h})$ ▷ Surface normal from heading in target pose estimate
12:         $s \leftarrow \left[q_{k_x}, q_{k_y}, q_{k_z}\right]^T + r_d \cdot \hat{\mathbf{n}}_{\mathbf{vk}}$ ▷ Calculate the desired extinguishing position
13:         $R_i, R_o \leftarrow$ CHECK_INNER_OUTER_RANGE(o_p, s) ▷ Check compliance with the ranges in Figure 12.
14:         **if** $H = false$ & $R_i = true$ **then**
15:             $H \leftarrow true$ ▷ Set pose hysteresis state to Floating
16:         **end if**
17:         **if** $H = true$ & $R_o = false$ **then**
18:             $H \leftarrow false$ ▷ Set pose hysteresis state to Controlling
19:         **end if**
20:         **if** IS_RECENT(t_{vv}) **then**   ▷ Fresh direction from RGB vision takes priority
21:             $\mathbf{d} \leftarrow \mathbf{v_v}$
22:             $e_z \leftarrow (v_{v_z} - z_z) \cdot r_d$
23:         **else if** IS_RECENT(t_{vt}) **then**   ▷ Thermal vision is more consistent but less precise
24:             $\mathbf{d} \leftarrow \mathbf{v_t}$
25:             $e_z \leftarrow (v_{t_z} - z_z) \cdot r_d$
26:         **else**   ▷ The pose estimate from Kalman filter is a fallback
27:             $\mathbf{d} \leftarrow \left[q_{k_x}, q_{k_y}, q_{k_z}\right]^T - \left[o_{p_x}, o_{p_y}, o_{p_z}\right]^T$
28:             $e_z \leftarrow d_z$
29:         **end if**
30:         $e_\eta \leftarrow$ GET_HEADING_DIFFERENCE$(\mathbf{z}, \mathbf{d})$ ▷ **z** is the direction of the water nozzle
31:         **if** $H = true$ **then**
32:             $n = 0$ ▷ Send only one trajectory point
33:             $r_0 \leftarrow \left[o_{p_x}, o_{p_y}, o_{p_z} + e_z\right]^T, \eta_0 \leftarrow o_{p_\eta} + e_\eta$ ▷ Control only the heading and altitude
34:             $W \leftarrow true$ ▷ Deploy water
35:         **else**
36:             $\mathbf{c} \leftarrow \left[s_x, s_y\right]^T - \left[o_{p_x}, o_{p_y}\right]^T$ ▷ Get horizontal offset from the desired extinguishing position
37:             $n \leftarrow \|\mathbf{c}\| / (v_{max} \cdot t_s)$ ▷ We need $n + 1$ trajectory points with time step of t_s
38:             $\mathbf{c_i} \leftarrow \mathbf{c} \cdot (v_{max} \cdot t_s / \|\mathbf{c}\|)$ ▷ We request returning speed of v_{max}
39:             **for** $i = 0$ to n **do**
40:                 $r_i \leftarrow \left[o_{p_x} + i \cdot c_{i_y}, o_{p_y} + i \cdot c_{i_x}, o_{p_z} + e_z\right]^T, \eta_i \leftarrow o_{p_\eta} + e_\eta$
41:             **end for**
42:             $W \leftarrow false$ ▷ If pose hysteresis state is not Floating, turn the water pump off
43:         **end if**
44:         $\mathcal{P}_{ref} \leftarrow \{(r_0, \eta_0), ..., (r_n, \eta_n)\}$
45:         SET_TRAJECTORY_REFERENCE$(\mathcal{P}_{ref})$ ▷ Send the trajectory points to the reference tracker
46:         **if** $W \neq W_p$ **then**   ▷ If the state has changed, turn the water pump on or off
47:             TOGGLE_WATER_PUMP(W)
48:         **end if**
49:         **if** $W = true$ **then**
50:             $w_t \leftarrow w_t - dt$ ▷ Keep track of the remaining time of water deployment
51:         **end if**
52:         SLEEP(dt) ▷ Enforce constant time period (0.1 s)
53:     **end while**
54:     TOGGLE_WATER_PUMP$(false)$
55:     YIELD_CONTROL_TO_STATE_MACHINE
56: **end function**

heading was necessary, otherwise the drifting would throw off the aim of the UAV. It is in this state that the UAV is allowed to spray water. The Z-coordinate and heading are controlled either to aim at the directly observed target (water collecting opening or significant flames, see section 6.1 and 6.2), or (if it is currently not visible *e.g.* due to being cooled down by the water) to aim at its estimated position from the Kalman filter. If the target is directly observed, the aim is more responsive to disturbances, while if the aim relies on the filter, the precision is lowered. Despite this, we have determined that for success of the mission it is better to spray water even when we do not have direct observation, since such observation was often lost and rapidly starting and stopping the water stream affected the precision more negatively than drifts in the filter. The UAV is only allowed to correct its X-Y coordinates again once it has been moved outside of the outer ranges (green zone in Figure 12), at which point spraying water is disabled. Tuning this behavior took a considerable amount of preparation time as its success was dependent on many variables not present in our simulation and testing was demanding in terms of time, personnel, and conditions. Apart from the actuated nozzle approach mentioned in section 4, a notable theoretical alternative to this approach - a paradigm that currently has the attention of the robotics and UAV communities (Rajappa et al., 2015; Convens et al., 2017; Tadokoro et al., 2017) - are fully-actuated platforms that allow for direct tilt control decoupled from horizontal translation. Such platforms are however highly experimental and we have thus decided for better-tested conventional platforms for this task.

7. Performance Results

7.1. Preparation

Before the competition officially began, the system was incrementally developed and rigorously tested. The preliminary testing and assembly of hardware components was done in Czech Republic. However, six weeks before the competition, the team relocated to a desert area in the United Arab Emirates. The reasons for this were twofold: the weather was unsuitable for flying in Czech Republic shortly before the competition due to it being winter. Our long-term experience says that the weather, lighting, and other local conditions have profound effects on the results of robotic deployment, especially in terms of sensor outputs, and properly preparing a robotic system to reliably operate in a certain region requires testing in the same region. For instance, the stronger sunlight and higher temperatures in UAE affected thermal imaging in ways that were not apparent in our central European country.

In UAE, we performed numerous tests for establishing the right approach to the task as described in the previous sections, as well as for tuning parameters. Initially, we performed tests with only the fire extinguishing subsystems on fire analogues (see Figure 15.). In these tests, we had shown that the aim was correct on an average of 93% of the time when performing the fire extinguishing actions (Section 6.4 (Spurny et al., 2021)), which shows the effectiveness of the 3D localization and extinguishing procedures in case of reliable target detection. In one of the latest stages of the development in the desert environment, we performed successful following of a wall, followed then by spraying water onto a fire analogue (this was captured on video[9] and samples are in Figure 16). The fire analogue was identical to fire analogues used in other parts of the fire challenge of the MBZIRC 2020 competition, since we could not safely use open flames. These were detected only with thermal vision in a significantly less complex mode where only an aluminium heating element was detected as a small concentrated spot of elevated temperature. However, these tests allowed us to develop and tune the fire extinguishing procedure described in section 6.4. The plot of the extinguishing trajectory from such a test, shown in Figure 16, clearly demonstrates compliance of the UAV motion with the motion hysteresis from Figure 12.

These real-world tests were interspersed with frequent testing in Gazebo simulation (see Figure 13). This type of simulation testing is a staple of modern robotics and is extremely useful for

---

[9] http://mrs.felk.cvut.cz/fr2020firechallenge-facadefires

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

Extinguishing real fires by fully autonomous multirotor UAVs in the *MBZIRC 2020* competition · **427**
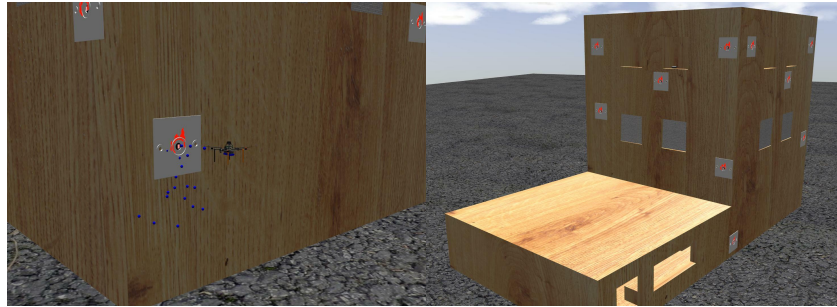
Figure 13. Extinguishing of facade fire objects and building overview in Gazebo simulator using our plugins.
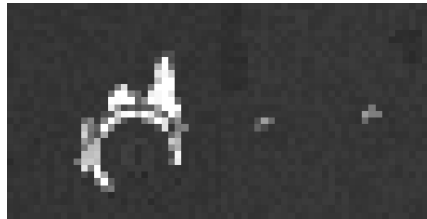
Figure 14. Outputs of the simulated version of our thermal cameras in Gazebo simulator.

Figure 15. Preliminary test in the desert with only fire extinguishing on a fire analogue.

saving resources and time when testing new changes to a system. In the simulator, in addition to emulating all the hardware onboard our *T650*-based UAVs, we have developed a plugin for emulating thermal cameras with outputs tuned to closely match the outputs of our real thermal cameras (see Figure 14), as well as the water spraying system. This software is open-source and is available online[10]. This allowed us to run the very same software, including the thermal vision, on the simulated UAVs as on the real UAVs, streamlining the development.

––––––––––

[10] https://github.com/ctu-mrs

428 · Walter et al.
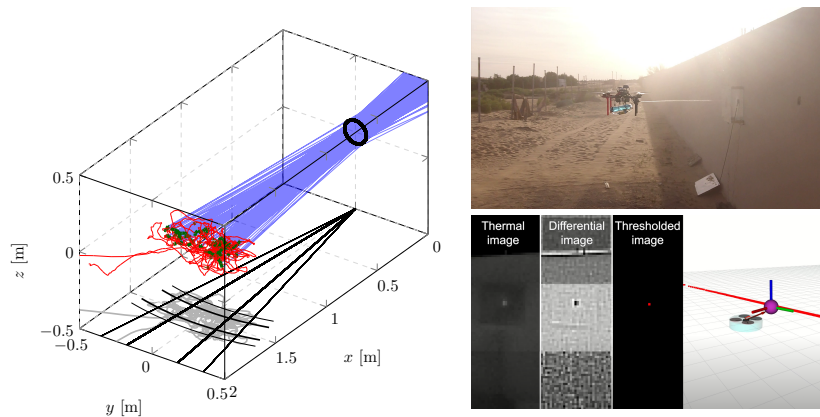
Figure 16. Extinguishing of fire analogue during our preparations in the desert. Green section of the trajectory on the left are points from which the water pump was activated, while blue lines represent the theoretical lines of water stream.
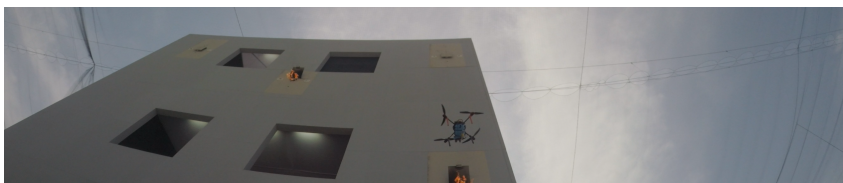
Figure 17. Bottom view of the UAV detecting a fire object during rehearsals of the competition.

7.2. Competition

Before the competition, we had been testing flights in the competition arena during rehearsal runs (Figure 17). This allowed us to check for potential negative effects of the environment, as well as to record datasets of the real fire objects. In the competition itself, we flew in two main trials of Challenge 3, as well as in the Grand Challenge of the competition and in the Final exhibition before the award ceremony. In the two rounds of trials, our team competed in the subtask of extinguishing the real fire objects described in this paper with two UAVs. We also competed in the subtasks of extinguishing ground fires with fire blankets with one UAV and extinguishing interior fires with a UGV.

Trial 1

In the first trial of the competition, one of our UAVs successfully sprayed water into one of the facade fire objects. This can be seen in Figures 18 and 11 - A. Unfortunately, this amount was very small and registered as 8 mL. This result was due to the conditions for spraying water being too strict in this phase. Specifically, the UAV was only permitted to spray water if both thermal and RGB vision provided fresh target localization to prevent wasting the limited capacity of carried water, as opposed to merely having active tracking from previous detection. This requirement was relaxed in subsequent runs to allow spraying in case of direct observation with either thermal vision or RGB vision, as long as the UAV was within the Floating phase of the pose hysteresis (see Algorithm 3).

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND
STABILIZATION TECHNIQUES

Extinguishing real fires by fully autonomous multirotor UAVs in the *MBZIRC 2020* competition · **429**

Figure 18.   Active extinguishing of a facade fire object in the first trial of the competition.

Additionally, water spraying was allowed regardless of the directional offset between the water nozzle and the estimated target direction. This meant that in the subsequent runs, water would more often be sprayed off-target, but the target was more likely to be hit with significantly more water (see subsection Final Exhibition below). In this trial, we also lost a significant amount of time due to high interference with the GNSS signal. Because of this, one of the water spraying UAVs did not take off after an emergency landing and another for more than 5 min. The primary culprit was the influence of sensors connected using USB 3.x. One of these devices onboard our UAVs was a *Realsense D435* camera. The precision of most GNSS localization systems can be severely affected by the components transmitting data via the cable at frequencies close to those used by GNSS. See sheet[11] for a detailed description of USB 3 frequency interference. However, we had observed these effects before and made some changes to our UAVs to mitigate them. Notably, we added multiple layers of aluminium foil between the GNSS and the rest of the components, but we could not fully overcome the severity of this interference. Our team did, however, successfully deploy over 1 L of water into a fire analogue using a UGV.

Trial 2

In the second trial, we did not successfully deploy water into the fire objects due to two separate incidents. The GNSS of one of the UAVs recorded a large sudden jump of position estimation while close to the building. Even though we have implemented a virtual repulsion force based on the LIDAR data, the jump was too large for this safeguard to overcome and the UAV hit the wall of the building in an attempt to correct its perceived erroneous position. The second incident occurred when the other UAV correctly detected and approached a fire object above the roof of a lower part of the building. The onboard laser altimeter of the UAV suddenly detected the change in perceived altitude when crossing over the edge of the roof. The UAV did not immediately crash, but started to oscillate in altitude which was exacerbated by the flexible textile lining of the roof oscillating due to the air pressure from the propellers of this UAV. Additionally, the fire object that the UAV approached had its air blowers active. Upon approach, the two combined sources of disturbance fatally destabilized the UAV, making it crash on top of the roof. Despite these problems, our team did score seven points with our blanket placing UAV.

While the total maximum possible score in Challenge 3 was 100 points, the achieved score of the victorious team was only 12.2626. This was followed by two teams tied at 10 points each, and our team tied in fourth place with another team at 7 points. The time left when a team declares the end of their turn was used as a rank tie-breaker when possible.

---

[11] https://www.intel.com/content/www/us/en/products/docs/io/universal-serial-bus/usb3-frequency-interference-paper.html - Accessed: 03-30-2022

430 · Walter et al.

Grand Challenge

The Grand Challenge of the competition consisted of performing selected subtasks of each of the three challenges of the MBZIRC 2020 in one time-slot of 25 min. We had prepared two UAVs for tasks from Challenge 1, one UAV for the task from Challenge 2, and one UGV and one UAV for tasks from Challenge 3 since our team had participated in all three challenges with full robotic autonomy in each. The last UAV was equipped to perform the liquid-based fire extinguishing described in this paper. In order to qualify for the financial award, a team had to score at least some points autonomously from each of the three reduced challenges. Unfortunately, we had to suspend deployment of the fire extinguishing UAV in favor of repeated attempts at successful scoring points in the reduced Challenge 2. At this point of planning, we had already scored 72 points in the reduced Challenge 1, as well as 12.5 points in the reduced Challenge 3 thanks to the success of the UGV. Only points from Challenge 2 were absent due to various complications with the associated UAV. Having the water-spraying UAV in air could interfere in various ways with this UAV, and therefore we recalled it shortly after takeoff and grounded it for the rest of the Grand Challenge. Overall, we did not score points in the reduced Challenge 2. Regardless, we ended up with the best total ranking in the Grand Challenge, making us the victorious team.

Final exhibition

After the Grand Challenge and before the award ceremony where the rankings were to be announced, the teams with the best results so far were offered the chance to perform a final exhibition of their systems. As our team qualified, we prepared one UGV and one UAV for repetition of Challenge 2, and one UGV and two UAVs for repetition of Challenge 3. The last two UAVs were once again prepared to do the task described in this paper and did so successfully in this last run. The UAVs found one facade fire object each and proceeded to deploy water into them, even simultaneously for a short time. Despite one of them hitting the wall at the end of this attempt due to the same type of GNSS jump as in Trial 2, by that point they deposited 56 mL and 61 mL respectively into the openings. Lastly, after the other UAV had depleted its water onto the fire object, it safely returned to the landing area without damage. This UAV is shown extinguishing fire during the final exhibition in Figures 11 - B and 20 with its trajectory in this attempt shown in Figure 19.

8. Beyond the MBZIRC and Future Work

Based on the lessons learned in MBZIRC, we are currently working on various improvements to the system for realistic firefighting in order to address the main drawbacks of the solution optimized for the competition trials. During the competition, it was also notable that of the two deployed fire extinguishing UAVs, one had a 3D-printed water nozzle made with less precision. This resulted in the side effect of its water stream being spread noticeably wider (see Figure 11) which was detrimental in terms of the competition. This has even lead to more water spillage outside of the measuring receptacle during preliminary testing. However, in practical terms this appears to be the superior design as the more spread-out water stream tended to extinguish the actual flames more consistently.

Another observation made during the competition concerns the nozzle aiming procedure. The nozzle, which was rigidly attached to the UAV body, made the aiming complex due to the need to use the already constrained degrees of freedom of the UAV. The fire extinguishing was less flexible than with a gimballed nozzle and was noticeably disturbed by drifting. We have previously addressed this issue by temporarily allowing the UAV to drift while only correcting the altitude and heading, but this solution solves the problem only partially in a real application. One potential solution to this problem would be to decouple the aiming from the UAV position control by using a gimballed nozzle actuated by servos based on target localization. If that had been the case, hovering in front of the fire objects would have been significantly smoother and the extinguishing would have become more

Extinguishing real fires by fully autonomous multirotor UAVs in the *MBZIRC 2020* competition · **431**
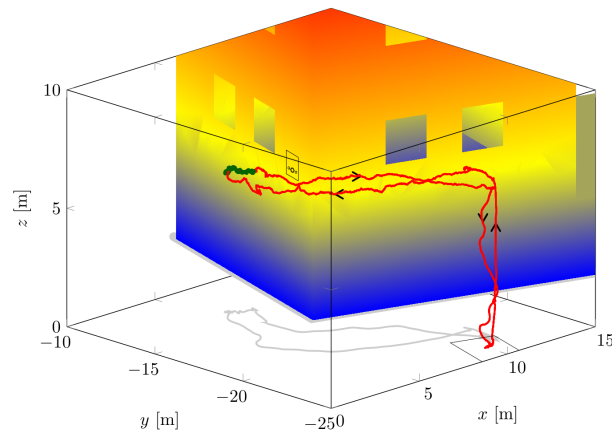
Figure 19. The trajectory of one of the extinguishing UAVs during the final exhibition, as recorded based on onboard GNSS receiver. The UAV took off at the designated area, circled around the building until it made a detection, and then localized a fire object. It then proceeded to approach it, spray water until the water was depleted, and then returned to the assumed position of the take-off area. The green section of the trajectory shows where the water pump was active. The notable drift during this time is difficult to analyze post-hoc and the most likely explanation for it is merely drift of the GNSS itself being dynamically compensated for by the sensor-based fire object position estimate. This also means that the shown position of the building is only approximate.
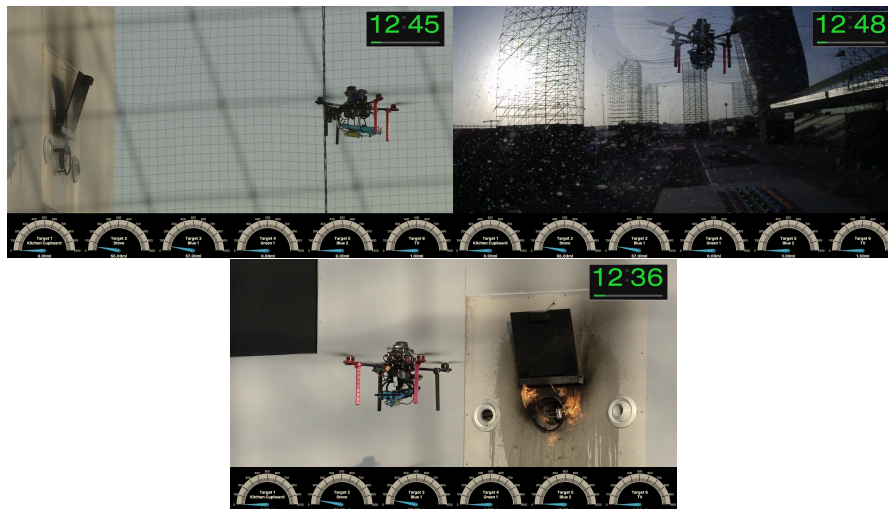
Figure 20. External views of the fire extinguishing in the final exhibition of MBZIRC 2020. The trajectory the UAV took is shown in Figure 19. The view from the onboard RGB camera for this flight is in Figure 11 - B and onboard thermal camera views from this flight are used in Figure 6.

432 · Walter et al.

Figure 21. Our working prototype of a firefighting UAV designed for using gas-based extinguishing capsule launcher - an evolution of the system developed for the MBZIRC 2020 competition.

consistent with fewer interruptions. We have not used this approach for the considerations discussed previously, but for a more stable UAV of larger scale equipped for a real firefighting deployment, this could also be a viable alternative to our approach, provided that the effects of the additional degrees of freedom in the system are properly studied.

However, we realized together with our industrial partners that spraying a liquid agent with a nozzle is altogether a sub-optimal approach to the task regardless, as it induces the need to involve a precisely-controlled UAV in situ for an extended period of time. Moreover, the firefighting agent needs to be dispersed throughout the entire inflamed room all at once to prevent re-ignition. To solve these requirements, we propose deploying capsules with a rigidly enclosed fire extinguishing agent into a localized fire using an onboard gas-based launcher. Such a solution makes it necessary to correctly aim using the thermal-based detection and servoing only for the instant before deployment of the capsule. The temporary disturbance that the launcher enacts on the UAV will not affect the firefighting effort itself. An overview of a prototype UAV with such a highly effective firefighting launching device can be seen in Figure 21.

9. Conclusion

A system designed for the fast extinguishing of fires on buildings by autonomous aerial robots equipped with onboard sensors and processing power was presented in this paper. The proposed solution enables scanning of a facade of a building while searching for hot regions indicating the presence of fire, localization of the fire source, and alignment of an onboard firefighting mechanism to precisely apply a liquid fire extinguishing agent. The novelty of the system lies in its onboard hybrid vision used for localizing facade fires, its capability of additionally observing surfaces behind flames, and the deployment of an extinguishing liquid based on body-pose-based dynamic aiming. In addition to the firefighting maneuvers, the system performs complex missions, takeoff to landing with full autonomy by exploiting onboard subsystems for UAV state estimation, localization, stabilization, and motion planning.

The fire localization and extinguishing systems of this solution are suitable for practical deployment, as was demonstrated during the MBZIRC 2020 competition that sought to emulate a real first-responders mission. According to the best of our knowledge, it was the only approach that achieved successful UAV-based firefighting of real fires in the MBZIRC 2020 without using RTK-GNSS or manual intervention - both of which were penalized in the competition as they do not follow the requirements of real-world applications. The presented solution was one of the key components of the system designed by our team - CTU-UPENN-NYU - that won the Grand challenge of the competition and was also successful in all three individual challenges. More importantly, the performance of the system in the MBZIRC 2020 competition motivated further research and development towards an industrial solution of this important task, as was the main goal of the MBZIRC 2020 organizers and its technical committee.

CHAPTER 3. IN-FLIGHT VISUAL RELATIVE LOCALIZATION AND STABILIZATION TECHNIQUES

Extinguishing real fires by fully autonomous multirotor UAVs in the *MBZIRC 2020* competition · **433**


Figure 22. Photos from the MBZIRC 2020 competition in Abu Dhabi, United Arab Emirates.

Besides the system design itself, this paper also documents several practical observations made during the development and the competition itself on the performance of the equipment used. We believe that these can be of additional value for the future development of similar systems. We have documented the need for computer vision that is robust to the highly dynamic appearance of fires in RGB and thermal vision, while highlighting the individual drawbacks of each. For RGB vision, we have pointed out how the transparency of flames to blue light can be useful for observing surfaces of interest hidden behind flames. Lastly, we have shown the need for robust aiming and localization that is resistant to various types of platform drifting and have documented more specific adverse effects, such as the interference of USB 3.x connected devices with the onboard GNSS receivers.

Acknowledgments

ORCID

Viktor Walter ⓘ https://orcid.org/0000-0001-8693-6261
Vojtěch Spurný ⓘ https://orcid.org/0000-0002-9019-1634
Matěj Petrlík ⓘ https://orcid.org/0000-0002-5337-9558
Tomáš Báča ⓘ https://orcid.org/0000-0001-9649-8277
David Žaitlík ⓘ https://orcid.org/0000-0001-9116-6265
Lyubomyr Demkiv ⓘ https://orcid.org/0000-0002-2802-3461
Martin Saska ⓘ https://orcid.org/0000-0001-7106-3816

References

Andraši, P., Radišić, T., Muštra, M., and Ivošević, J. (2017). Night-time detection of uavs using thermal infrared camera. *Transportation Research Procedia*, 28:183–190.

434 · Walter et al.

Ashour, R., Taha, T., Mohamed, F., Hableel, E., Kheil, Y., Elsalamouny, M., Kadadha, M., Rangan, K., Dias, J., Seneviratne, L., and Cai, G. (2016). Site inspection drone: A solution for inspecting and regulating construction sites. pages 1–4.

Baca, 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. In *IEEE IROS*, pages 1–8.

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.

Cazzato, D., Cimarelli, C., Sanchez-Lopez, J., Voos, H., and Leo, M. (2020). A Survey of Computer Vision Methods for 2D Object Detection from Unmanned Aerial Vehicles. *Journal of Imaging*, 6:78.

CBInsight (2020). 38 Ways Drones Will Impact Society: From Fighting War To Forecasting Weather, UAVs Change Everything. Available: https://www.cbinsights.com/research/drone-impact-society-uav/ (Accessed: 01-09-2020).

Chen, R., Cao, H., Cheng, H., and Xie, J. (2019a). Study on Urban Emergency Firefighting Flying Robots Based on UAV. In *IEEE IAEAC*, volume 1, pages 1890–1893.

Chen, T., Yuen, A. C. Y., Yeoh, G., Yang, W., and Chan, Q. (2019b). Fire risk assessment of combustible exterior cladding using a collective numerical database. *Fire*, 2:11.

Convens, B., Merckaert, K., Nicotra, M. M., Naldi, R., and Garone, E. (2017). Control of fully actuated unmanned aerial vehicles with actuator saturation. *IFAC-PapersOnLine*, 50(1):12715–12720.

Emicontrols (2018a). 3 new taf20 firefighting robots for shanghait. Available: https://www.emicontrols.com/en/stories/3-new-taf20-firefighting-robots-for-shanghait-16/ (Accessed: 06-17-2020).

Emicontrols (2018b). Taf35 firefighting robot – for more safety when fighting fires. Available: https://www.emicontrols.com/en/fire-fighting/mobile-fire-protection/firefighting-robot-taf35 (Accessed: 06-17-2020).

Fire Product Search (2018). New fire ox can do what humans can't. Available: https://www.fireproductsearch.com/new-fire-ox-can-humans-cant/ (Accessed: 06-17-2020).

Firehouse (2018). Product of the day: Thermite firefighting robot. Available: https://www.firehouse.com/operations-training/hoselines-water-appliances/product/21142939/howe-howe-technologies-inc-thermite-firefighting-robot (Accessed: 06-17-2020).

Goecks, V. G., Woods, G., and Valasek, J. (2020). Combining visible and infrared spectrum imagery using machine learning for small unmanned aerial system detection. 11394:113940Z.

Guillaume, E., Fateh, T., Ukleja, S., Chiva, R., and Schillinger, R. (2018). Study of fire behaviour of facade mock-ups equipped with aluminium composite material-based claddings, using intermediate-scale test method. *Fire and Materials*.

Hopkins, M. A., Griffin, R. J., Leonessa, A., Lattimer, B. Y., and Furukawa, T. (2015). Design of a compliant bipedal walking controller for the DARPA Robotics Challenge. In *IEEE-RAS International Conference on Humanoid Robots*, pages 831–837.

Hrabia, C.-E., Hessler, A., Xu, Y., Seibert, J., Brehmer, J., and Albayrak, S. (2019). Efffeu project: Towards mission-guided application of drones in safety and security environments. *Sensors*, 19:973.

Hubbard, B. and Hubbard, S. (2020). Unmanned Aircraft Systems (UAS) for Bridge Inspection Safety. *Drones*, 4:40.

Lattimer, B. Y. (2020). Robotics in firefighting. Available: https://www.sfpe.org/page/FPE_ET_Issue_100/Robotics-in-Firefighting.htm (Accessed: 06-17-2020).

Lee, S. M., Chien, J. L., Tang, E., Lee, D., Liu, J., Lim, R., and Foong, S. (2020). Hybrid kinematics modelling for an aerial robot with visual controllable fluid ejection. In *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 832–838.

Lee, T., Leoky, M., and McClamroch, N. H. (2010). Geometric tracking control of a quadrotor UAV on SE(3). In *IEEE CDC*, pages 5420–5425.

Liljeback, P., Stavdahl, O., and Beitnes, A. (2006). Snakefighter - development of a water hydraulic fire fighting snake robot. In *2006 9th International Conference on Control, Automation, Robotics and Vision*, pages 1–6.

London Fire Brigade (2018). London fire brigade operational response to grenfell tower. Available: https://www.grenfelltowerinquiry.org.uk/evidence/london-fire-brigade-operational-response (Accessed: 01-09-2020).

Extinguishing real fires by fully autonomous multirotor UAVs in the *MBZIRC 2020* competition · **435**

McKenna, S., Jones, N., Peck, G., Dickens, K., Pawelec, W., Oradei, S., Harris, S., Stec, A., and Hull, R. (2018). Fire behaviour of modern façade materials – understanding the grenfell tower fire. *Journal of Hazardous Materials*, 368.

Measure (2016). Drones for power plants. Available: https://www.measure.com/hubfs/PDF%20Resources/ Drones_PowerPlants.pdf (Accessed: 06-17-2020).

Merino, L., Caballero, F., Martinez-de Dios, J. R., Maza, I., and Ollero, A. (2012). An Unmanned Aircraft System for Automatic Forest Fire Monitoring and Measurement. *Journal of Intelligent and Robotic Systems*, 65:533–548.

Minkina, W. and Dudzik, S. (2009). *Infrared thermography: errors and uncertainties*. John Wiley & Sons.

NFPA, N. F. P. A. (2019). U.s. fire statistics.

Nousi, P., Mademlis, I., Karakostas, I., Tefas, A., and Pitas, I. (2019). Embedded UAV Real-Time Visual Object Detection and Tracking. In *IEEE RCAR*, pages 708–713.

Pecho, P., Magdolenová, P., and Bugaj, M. (2019). Unmanned aerial vehicle technology in the process of early fire localization of buildings. *Transportation Research Procedia*, 40:461–468.

Peskoe-Yang, L. (2019). Paris firefighters used this remote-controlled robot to extinguish the notre dame blaze. Available: https://spectrum.ieee.org/automaton/robotics/industrial-robots/colossus-the-firefighting-robot-that-helped-save-notre-dame (Accessed: 06-17-2020).

Portas, J., Bergesio, L., Campaña, I., Vaquero-Melchor, D., López-Araquistain, J., Barbolla, A., and Casar, J. (2018). Drone mission definition and implementation for automated infrastructure inspection using airborne sensors. *Sensors*, 18:1170.

Qin, H., Cui, J. Q., Li, J., Bi, Y., Lan, M., Shan, M., Liu, W., Wang, K., Lin, F., Zhang, Y. F., and Chen, B. M. (2016). Design and implementation of an unmanned aerial vehicle for autonomous firefighting missions. In *IEEE ICCA*, pages 62–67.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A. Y. (2009). ROS: an open-source Robot Operating System. In *IEEE ICRA workshop on open source software*, volume 3, page 5.

Rajappa, S., Ryll, M., Bülthoff, H. H., and Franchi, A. (2015). Modeling, control and design optimization for a fully-actuated hexarotor aerial vehicle with tilted propellers. In *IEEE ICRA*, pages 4006–4013.

Rangan, M. K., Rakesh, S. M., Sandeep, G. S. P., and Suttur, C. S. (2013). A computer vision based approach for detection of fire and direction control for enhanced operation of fire fighting robot. In *2013 International Conference on Control, Automation, Robotics and Embedded Systems*, pages 1–6.

Restas, A. (2006). Forest Fire Management Supporting by UAV Based Air Reconnaissance Results of Szendro Fire Department, Hungary. In *IEEE ISEIMA*, pages 73–77.

Ritchie, H. (2018). Causes of death. *Our World in Data*. Available: https://ourworldindata.org/causes-of-death (Accessed: 06-17-2020).

Robotpompier (2018). Tc800-ff technical assistance and fire-fighting robot. Available: https://www. robotpompier.com/en/ (Accessed: 06-17-2020).

Safetymanagement (2018). The use of robotics in firefighting. Available: https://safetymanagement.eku. edu/blog/the-use-of-robotics-in-firefighting/ (Accessed: 06-17-2020).

Saikin, D. A., Baca, T., Gurtner, M., and Saska, M. (2020). Wildfire fighting by unmanned aerial system exploiting its time-varying mass. *IEEE Robotics and Automation Letters*, 5(2):2674–2681.

Schatsky, D. and Ream, J. (2016). Drones mean business. Available: https://www2.deloitte.com/content/ dam/Deloitte/ie/Documents/Technology/DUP_Signals-for-Strategists_Drones-mean-business.pdf (Accessed: 06-17-2020).

Schulte, S., Hillen, F., and Prinz, T. (2017). Analysis of Combined Uav-Based Rgb and Thermal Remote Sensing Data: a New Approach To Crowd Monitoring. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42:347.

Science|Business (2020). Robots flex their firefighting skills. Available: https://sciencebusiness.net/robots-flex-their-firefighting-skills (Accessed: 06-17-2020).

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.

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.

Tadokoro, Y., Ibuki, T., and Sampei, M. (2017). Maneuverability analysis of a fully-actuated hexrotor uav considering tilt angles and arrangement of rotors. *IFAC-PapersOnLine*, 50(1):8981–8986.

436 · Walter et al.

The Verge (2019). Dji drones helped track and stop the notre dame fire. Available: https://www.theverge.com/2019/4/16/18410723/notre-dame-fire-dji-drones-tracking-stopped-thermal-cameras (Accessed: 10-06-2021).

Tudevdagva, U., Battseren, B., and Hardt, W. (2017). Unmanned aerial vehicle based automated inspection system for high voltage transmission lines. volume 1, pages 28–32.

USFA, U. F. A. (2018). Fire loss in the united states during 2018.

Vidas, S., Moghadam, P., and Bosse, M. (2013). 3D Thermal Mapping of Building Interiors using an RGB-D and Thermal Camera. *IEEE ICRA*, pages 2303–2310.

Viguria, A., Maza, I., and Ollero, A. (2010). Distributed Service-Based Cooperation in Aerial/Ground Robot Teams Applied to Fire Detection and Extinguishing Missions. *Advanced Robotics*, 24(1-2):1–23.

Vrba, M., Heřt, D., and Saska, M. (2019). Onboard marker-less detection and localization of non-cooperating drones for their safe interception by an autonomous aerial system. *IEEE Robotics and Automation Letters*, 4(4):3402–3409.

VT (2018). Trec: Terrestrial robotics engineering and controls. Available: http://www.me.vt.edu/research/laboratories/trec/ (Accessed: 06-17-2020).

WHO (2020). Global health estimates: Leading causes of death. Available: https://www.who.int/data/gho/data/themes/mortality-and-global-health-estimates/ghe-leading-causes-of-death (Accessed: 10-06-2021).

Extinguishing of Ground Fires by Fully Autonomous UAVs motivated by the *MBZIRC 2020* Competition

Viktor Walter[*], Vojtěch Spurný[*], Matěj Petrlík[*], Tomáš Báča[*], David Žaitlík[*] and Martin Saska[*]

Abstract—In this paper, a system for autonomous extinguishing of ground fires using the placement of fire blankets by Multi-rotor Unmanned Aerial Vehicles (UAVs) is proposed. The proposed system, relying on the fusion of multiple onboard sensors using only onboard computers, is infrastructure independent with a focus on high reliability in safety-critical missions that require power-on-and-go full autonomy. This task was part of the third challenge of MBZIRC 2020 aimed at the development of autonomous robotic systems for extinguishing fires inside and outside of buildings. The MBZIRC competition promotes the development of such robotics applications that are highly demanded by society and, due to their complexity and required robot abilities, go beyond the current robotic state of the art. As far as we are aware, our team was one of only two teams to achieve successful system for placement of fire blankets fully autonomously with vision-based target localization without using Real-time kinematic (RTK)-global navigation satellite system (GNSS), as was required in the competition and also for the real missions of first responders.

I. INTRODUCTION

The task of protecting life and property is at the forefront of robotics research. However, realistic deployment of autonomous aerial robotics systems for physical interaction in such a task still seems to be a matter for the distant future. Systems that demonstrate the required cognitive capabilities are almost exclusively confined to laboratory settings. This is due to the numerous difficult challenges for robotic systems interacting with real-world hazardous conditions, especially when health or lives are at stake.

In first responder missions, deployed units must have high speed and mobility in order to reach remote or inaccessible areas quickly. This is the primary strength of Multi-rotor Unmanned Aerial Vehicle (UAV)s which are finding an increasing number of applications in recent years. However, their limited payload makes using them for firefighting with standard liquid fire extinguisher a challenging endeavor. One of the most practical options for how these systems can be deployed is to use them to smother ground fires with fire blankets. Such fire blankets are a very effective and non-destructive fire extinguishing method (see Fig. 2[1]) that is

[*]Faculty of Electrical Engineering, CTU in Prague, Technická 2, Prague 6, `{viktor.walter|vojtech.spurny|matej.petrlik| tomas.baca|david.zaitlik|martin.saska}@fel.cvut.cz`

   [1]`pozarni-zbozi.cz/eshop-car-fire-blanket-haseni-horicich-automobilu -bez-pouziti-vody.html`
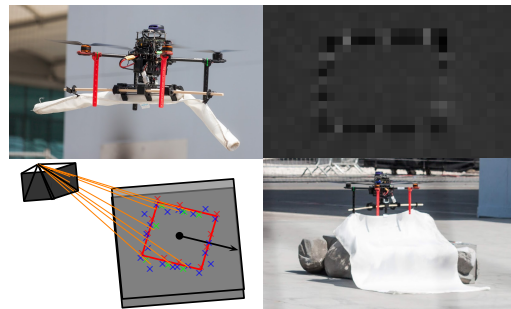
Fig. 1: The proposed system is based on target detection and pose estimation using thermal computer vision and a blanket unrolling maneuver informed by the retrieved pose of the target.

Fig. 2: Fire blanket used to extinguish a burning automobile. The footage was used as motivation for designing the firefighting maneuver presented in this paper. This task can now be accomplished autonomously by an upscaled version of the proposed system.

well suited for execution by Multi-rotor UAVs, as blankets are much lighter than the amount of liquid agent sufficient for extinguishment and they additionally prevent further re-ignition of flames where they are deployed. Challenge 3 of the Mohamed Bin Zayed International Robotics Challenge (MBZIRC) 2020 competition was inspired by the need to extinguish fires in locations that are difficult to access by people. A subtask of the challenge was directly motivated by deploying fire blankets on outdoor ground fires by a fully autonomous UAV system. This paper describes our approach to successfully fulfill this task in the competition, as well as in the real missions discussed. The proposed approach is based on using a combination of new thermal vision, perception and estimation subsystem, and a blanket unrolling procedure coupled with a proposed spreading maneuver inspired by real firefighting missions - see Fig. 1 and 2 for illustration. To our best knowledge, in this subtask our CTU-UPENN-NYU team was one of only two successful teams to use full autonomy in flight and to rely solely on onboard sensors for localization of the targets, while also not using RTK-GNSS or pre-measured target positions. Reliance

on the latter especially would render our system completely impractical for our main focus of future application in real fire-fighting, since if an operator had the option to physically access the fire for measurement of its coordinates, then using Multi-rotor UAVs would not be necessary.

II. STATE OF THE ART

The idea of using UAVs to support firefighting has already been extensively discussed in previous research. The simplest situation where UAVs can be applied is outdoor fire detection and monitoring. A system of multiple UAVs was used in [1] for automatic forest fire monitoring using visual and infrared cameras. Real experiments with forest fire monitoring in a national park have already been conducted by the Hungarian fire department [2]. The authors of [3] describe a task allocation strategy for distributed cooperation of ground and aerial robot teams in fire detection and extinguishing. In [4], a UAV system is designed for the delivery of fire-extinguishing bombs to a target area on remote control, using various internal and external data sources. Similarly, in [5] the authors employ a UAV system for delivering liquid fire-extinguishing agent by an extreme dynamic dropping maneuver maximizing the release velocity to reduce dispersion. However, this method of firefighting is only suitable for use in open space settings such as desert fires, without any tall or vulnerable structures. The development of firefighting UAVs has already been a topic of robotic competitions as well. [6] describes the design and implementation of a firefighting UAV for outdoor applications designed specifically for the IMAV 2015 competition. This system is based on collecting water into a carried receptacle and pouring it on a ground target - a solution that is of interest for research, but in practice only effective for vehicles with high payload, such as full-scale airplanes and helicopters. The employment of UAVs could prove particularly beneficial and life-saving in urban environments. Experiments have already been done on urban fire detection using a thermal camera which could be carried by a UAV [7]. UAVs capable of entering buildings through doors and windows would then be especially helpful with their ability to reach the target location much earlier than human firefighters. [8] contains the design of a semi-autonomous indoor firefighting UAV. The authors have designed a fireproof, thermoelectrically cooled UAV equipped with visual and thermal cameras, collision avoidance, and first person view system. However, the UAV was controlled remotely and no autonomy was demonstrated here, making it dependent on high-bandwidth radio transmissions and a fully engaged remote pilot. The solution discussed in this paper - placing fire blankets on top of a detected target - requires the ability of Multi-rotor UAVs to autonomously carry and deploy objects while precisely navigating towards a detected target. Carrying and dropping objects has been extensively researched in recent literature [9]–[11]. Object transportation by Multi-rotor UAVs [12], [13], as well as precise navigation of Multi-rotor UAV towards a visually localized target [14], has even been a task in MBZIRC 2017 where teams - including ours - competed in gathering of scattered objects and landing on a moving vehicle using vision-based tracking.

However, in comparison to the previous work, the approach proposed in this paper requires additional new capabilities. The UAV is required to precisely localize a target by fusing thermal imaging-based computer vision instead of simplistic color detection. Additionally, instead of merely carrying and detaching over a designated area, the transported blanket has to be deployed in multiple steps on an object of comparable size. In doing so, the system has to not only address, but actually exploit physical interaction with the ground and the target - a challenge rarely addressed in practical deployment of Multi-rotor UAVs.

III. TASK SPECIFICATION

Challenge 3 of the MBZIRC comprised four distinct parts: extinguishing real fires on the exterior of a building by Multi-rotor UAVs using water, extinguishing fire analogues using water in the interior of a building by Multi-rotor UAVs and by Unmanned Ground Vehicle (UGV), and extinguishing free-standing fire analogues by Multi-rotor UAV with fire blankets. This paper presents a solution for firefighting by fire blanket placement.

The fire analogues for the competition (see Fig. 5) were designed as square platforms with $1\,\mathrm{m}$ sides and $0.35\,\mathrm{m}$ in height. All edges of the top surface were framed with reflective metal sheets $0.1\,\mathrm{m}$ wide. Each side of the empty square formed by these sheets was studded with three electrical heating elements made of anodized aluminium - twelve in total - with one in the center of the edge and two spread $0.25\,\mathrm{m}$ to each side. When active, each of the elements was set to maintain a temperature of $120\,°\mathrm{C}$. The center of the square surface additionally contained a silkflame with a series of LED light sources in its center. Three of these fire analogues were spread out in an area of $19.5{\times}17.5\,\mathrm{m}$ and two of them were randomly switched on for each trial. The goal was to locate these objects with a Multi-rotor UAV and to cover the whole top surface of the fire analogue objects with a standard fire blanket 1.2×1.2 m in size. For each placement, a specific number of points are awarded for each covered corner and side as follows:

- Covering the entire top surface of the fire analogue by the blanket awards a team 10 points.
- Covering any part of the top surface of a fire analogue is worth $20\,\%$ of the total = 2 points.
- Covering any of the four corners or overlaying any of the four sides is worth $10\,\% = 1$ point.

All teams scoring in autonomous mode are ranked above all teams solving manually, reflecting the significantly higher difficulty and relevance of autonomous solutions. Additionally, teams were allowed to use RTK-GNSS in any of the competition arenas, but a penalty of $25\,\%$ was applied to teams if they did since the solution would be less relevant to real firefighting in inaccessible areas without the option of establishing a calibrated local base station. In such a case, the teams would have the option of pre-measuring the positions of the targets, lowering the need for sensing-based navigation

Fig. 3: Front view of the blanket dropping gripper below the body of the UAV (left) and the layout of the thermal cameras (right).

and target localization - the main focus of this paper and key component of a real system.

IV. PLATFORM DESCRIPTION

Our UAV for the blanket dropping challenge was based on the *Tarot T650* platform, equipped with *Intel NUC* onboard computer and the *Pixhawk* flight controller. On the onboard computer, the UAV runs a multi-layered system stack developed by our group [15], that allows optimal following of pre-defined and dynamically generated trajectories using Model predictive control (MPC). For odometry, the UAV was also equipped with a GNSS module with *Ublox Neo-M8N* receiver. The sensory equipment of the UAV comprised *RPLIDAR* rotary rangefinder, *Garmin* laser rangefinder used as altimeter, *BlueFOX-MLC200wC* camera used for calculation of optical flow from the ground, and two *Teraranger EVO thermal 33* thermal cameras with a field of view (FoV) of $33° \times 33°$. One of the thermal cameras was pointed downwards and the other was tilted $30°$ upwards from the first, facing towards the front of the UAV (see Fig. 3 on the right). The two cameras could then be used in place of a single camera with increased resolution and FoV. The downward-facing thermal camera was useful for getting precise estimates, while the front-facing one improved the capability of searching for undiscovered fire analogues by extending visibility further outwards.

This modular approach of using multiple simpler cameras was also beneficial in our work on two different types of Multi-rotor UAVs for Challenge 3. For example, we attached three thermal cameras of the same type onto a Multi-rotor UAVs meant for extinguishing fires on vertical surfaces with a water stream. Thus, we could address the diverging needs of observation coverage between the two tasks without the need for different sensors or working with different input data. Such modularity is also beneficial for real deployment as a single platform can be used for various firefighting tasks. For dropping the blanket, we developed a specialized mechanized claw that carries a rolled up fire blanket, one side of which is studded with a wooden rod. This mechanism releases the blanket in two phases. First, one servo releases the rolled up blanket such that it unfolds and remains attached to the UAV only by one side by the attached wooden rod. In the second phase, the wooden rod itself is released by the second part of the servomechanism, concluding the deployment. The system used two *Dynamixel AX-12A* servomotors.
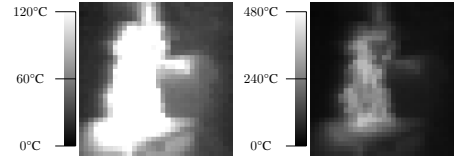
Fig. 4: An example of the view of a real fire with the thermal camera used in the competition at two different thermal ranges. Note the high contrast of the fire compared to the background, in addition to the large size of the fires in the image. In the range of temperatures used in the competition, the fire itself is completely saturated in the image. Detection and targeting of such objects is significantly easier to achieve than the fire analogues.

V. VISION

The output from the thermal cameras comprises a matrix of 32×32 pixels each, where the value of a pixel is supposed to represent the temperature of the corresponding region in degrees Celsius. This is, however, an oversimplification. The thermal cameras capture the intensity of infrared (IR) radiation. This intensity is converted using the Stefan-Boltzman law into a temperature reading, presuming a surface emissivity of 0.95 [16]. Such value is similar to the emissivity of a wide range of materials, such as iron or plastic, but differs significantly with other materials such as aluminium which has significantly lower emissivity, leading to lower temperature readings. Additionally, if the material in question is highly reflective in the given IR range, the thermal camera output in that region will represent properties of the reflected surfaces instead of the reflective material itself, akin to a mirror. In our experiments, as a real fire (see Fig. 4) is evaluated by a thermal camera as being over $250°$C, the thermal cameras would be sufficient to detect real fires from a significant distance and with good reliability, because such hot objects will likely be detected and will not be outliers.

Unfortunately, the heating elements used in MBZIRC were made of anodized aluminium, with emissivity index of approximately 0.55 [17]. Therefore, detection of the heating elements set to maintain a temperature of $120°$C was significantly more complicated, since the cameras parse them as having approximately only $78°$C when viewed from up close. Each element was a rectangle 60×35 mm. This small size combined with the small resolution compared to the FoV of the applied thermal cameras, as well as the typical searching distance of *approx.* 3 m above ground level, made it often project as less than a full pixel in the image. This effect further decreased the temperature read out of the pixels containing the heating elements, as the camera effectively averages the temperatures of the entire surface captured within a single pixel. In relatively cold and shaded surroundings, these elements may be distinguishable as concentrated patches of elevated temperature. However, in the real-world conditions of the competition (desert climate and sun-heated ground) these were barely noticeable, which would not be the case for real fires (compare Fig. 4 with
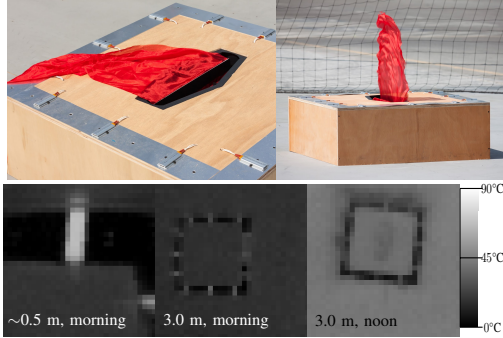
Fig. 5: Views of a fire analogue from a color camera and a thermal camera. In the later stages, the wooden sides were lined with rock facsimiles. Note that while up-close, the heating elements are recognizable as being measured with greater temperate than the surroundings (measured as *approx.* $78\,°C$ in the left image), from $3\,m$ above the object where the whole square is captured, their contrast w.r.t. the surroundings decreases significantly. This decrease is so severe that in the right image, captured at noon when the sun was directly above the area, the heating elements are indistinguishable based on temperature measurement alone as the temperature of the ground itself was $55\,°C$. Even in the middle thermal image captured in the morning, the brightest pixels correspond to a measurement of $40\,°C$ - only $10\,°C$ more than the surroundings. In contrast, the pixels corresponding to the metal plates have a value of $0\,°C$ when captured up close and $10\,°C$ from farther above for both cases shown here.

Fig. 5). In our preliminary tests and as was subsequently confirmed in the rehearsals with the fire analogues, we discovered that the reflective metal sheets underlying the heating elements themselves appeared as extremely cold (*approx.* $0\,°C$). This was due to their high IR reflectivity, making them reflect the sky instead of emitting their own detectable radiation.

For brevity, we will be calling pixels with such temperatures as "dark". It is significantly more efficient to detect these reflective metal sheets due to their high contrast with the surrounding surfaces and relatively large surface area. While it is assumed that these plates were meant to spread the heat of the elements over a larger area for detection, in terms of the thermal image they did the very opposite. We have exploited this fact to achieve sensitivity in detection comparable to that of real fires.

The metal sheets were segmented into smaller sections by placement of the heating elements that were without such reflectivity. This allowed us to detect the position and precise orientation of the fire analogues by the distinct square arrangement of dark patches on the edges of the fire analogues. Refer to Fig. 5 for the appearance of the fire analogues in the thermal camera images.

A. Shape fitting

The fire blanket required to be used in the competition had the dimensions 1.2×1.2 m. This blanket needed to

(a) Image processing

(b) Object point calculation   (c) Accumulation of points   (d) Target pose estimation
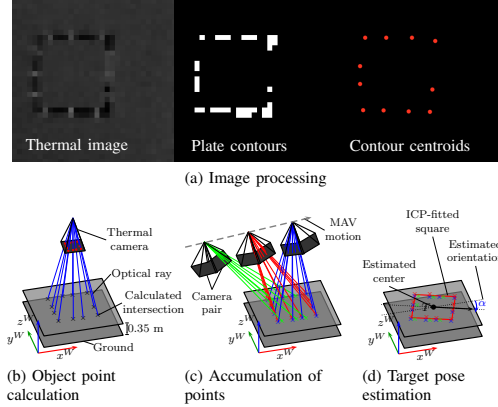
Fig. 6: The detection and localization pipeline for the blanket deployment subtask.

cover a fire analogue platform with the dimensions of 1.0×1.0 m. Therefore, it became necessary to estimate not only the precise position of the center of the platform, but also its orientation in order to align our UAV with its sides and cover as large a portion of the platform as possible. As mentioned above, the thermal camera captured the metal frame of the top surface of the platform as a set of dark patches arranged in a square. We took the image centroid of each of these patches (Fig. 6a) and estimated a 3D point it corresponds to. This 3D point is the centroid of the segment of the metal sheet defined by intersecting its corresponding optical ray with the estimated top surface plane of the platform (Fig. 6b).

The optical ray - the line defined by the coordinates of the camera center and a direction vector $\mathbf{v_t}$ - is obtained using odometry and the image coordinates of the centroid at x, y, combined with the assumed camera matrix for the used thermal cameras with pixel resolution w per side and FoV ϵ per side:

$$\mathbf{v_t} = \begin{bmatrix} 1/f & 0 & -((w-1)/2)/f \\ 0 & 1/f & -((w-1)/2)/f \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad (1)$$

where

$$f = \frac{(w/2)}{\tan(\epsilon/2)} \approx 54.015. \quad (2)$$

In 3D, the detections from both cameras that are oriented differently w.r.t. each other were translated using their known poses within the UAV frame into a unified set of points in a single coordinate frame. These 3D points were accumulated from all detections from both thermal cameras from the last $3\,s$ (Fig. 6c) and were then used for the target pose estimation step.

It should be noted that the primary source of the height estimate of our UAV w.r.t. the ground was a laser altimeter.
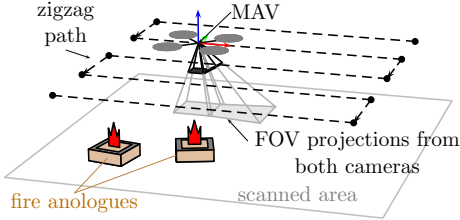
Fig. 7: Boustrophedon coverage of the competition arena.

Fig. 8: Diagram of the mission state machine.

Therefore, if the UAV is presumed not to be above the platform, we shift the intersecting plane by $0.35\,\mathrm{m}$ upwards to account for the height of the platform. While this was only possible to know after the pose of the target had been estimated, in practice this method quickly led to convergence to the correct pose. This is because, even with offset of the height of the platform, the estimated points form roughly the same shape, only scaled up and distorted when accumulating them from a time segment when the UAV transitioned above the target.

The next step involved applying a 2D Iterative Closest Point (ICP) algorithm to fit a square (Fig. 6d) of known size (0.9×0.9 m, accounting for the width of the metal frame) onto the accumulated points. The specific implementation of the ICP used was based on minimizing the sum of squares of distances of estimated points from the line segments representing the sides of the square.

The best precision of this estimation technique can be achieved when the UAV observes the platform from directly above. This is due to the ability to view the whole square shape in a single image frame, increasing the likelihood of seeing the dark reflections mentioned in section V and the negation of effects in this specific viewpoint, such as foreshortening and height estimate error.

This estimate is then sent to a linear Kalman filter to refine precision using multiple detections.

The state $\boldsymbol{q_k}$ of the Kalman filter comprises the 3D position of the platform surface (world frame in meters), as well as its 2D orientation q_{k_η} (azimuth in radians):

$$\boldsymbol{q_k} = \left[q_{k_x}, q_{k_y}, q_{k_z}, q_{k_\eta}\right]^T. \qquad (3)$$

In the case that new measurement α is close to 90° from the q_{k_η}, we rotate α by 90° towards q_{k_η}, once again taking into account the symmetry of the platform. In this way, the filter state is seamlessly updated even in cases when the target platform is rotated diagonally w.r.t. the world axes, where α would tend to oscillate between two expressions of the same pose and thus would otherwise corrupt q_{k_η}.

VI. SEARCH FOR FIRE ANALOGUES

A diagram of the state machine is shown in Fig. 8. The path planning for localization of the fire analogues in a predefined area can be described as Coverage Path Planning (CPP) [18]. The plan is found using Boustrophedon coverage [19], which creates a zigzag path as shown in Fig. 7, such that the reduced FoV of onboard thermal cameras entirely covers the area from the height of $3\,\mathrm{m}$. The reduced FoV is calculated based on the required overlap in the coverage (set to 5% during the competition) and on the camera FoV projected from flight height to the ground plane. The UAV then follows this path with heading in the direction of its movement while trying to detect and estimate the position of the fire analogues.
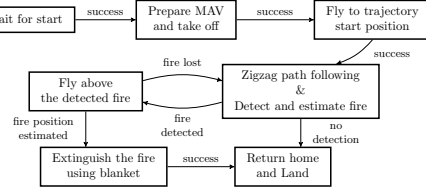
After successful detection, the UAV stops following the path and moves to a position $3.5\,\mathrm{m}$ above the estimated position of the fire analogue to better observe its pose. As the horizontal distance decreases, the visibility of the target increases and thus the estimate is improved. The UAV steers above the center of the target, where it hovers until the estimate reaches a desired precision. This requirement is evaluated based on the state covariance matrix $\boldsymbol{\Sigma}$ of the Kalman filter. The sufficiency of the precision of the position estimation is determined by the norm of eigenvalues of $\boldsymbol{\Sigma_{33}}$ - the 3×3 upper left submatrix of Σ representing the 3D position coordinates - and the last element of $\boldsymbol{\Sigma}$ representing the heading. The estimation is thus concluded once

$$\lambda_X = \|\lambda(\boldsymbol{\Sigma_{33}})\| \qquad < 0.1\,\mathrm{m} \qquad (4)$$
$$\lambda_\eta = \Sigma_{44} \qquad < (\pi/180) \cdot 5°. \qquad (5)$$

If λ_X and λ_η is smaller than such values, it is an indicator that multiple consistent measurements have contributed to the filter state $\boldsymbol{q_k}$ and thus $\boldsymbol{q_k}$ represents a reliable-enough estimate for blanket placement.

Once the desired precision is achieved, the blanket placement maneuver is initiated. In the case of no detection, the UAV flies back to the takeoff position to land, as would have been the case if the blanket placement was successful. If the target was detected but subsequently lost, the UAV continues the path following from the last visited point.

VII. PLACEMENT MANEUVER

Upon reaching the desired estimation precision, the UAV activates its blanket placement maneuver. This begins with selecting a direction from which to place the blanket. Since the maneuver entails flight parallel to the sides of the square platform for optimal coverage, there are four options for the starting position. These positions are all $4\,\mathrm{m}$ from the center of the estimated target and positioned along the estimated

Fig. 9: Our UAV approaching the fire analogue in the placement maneuver. The fire blanket is unrolled when the UAV is $3\,\mathrm{m}$ from the target. The second end of the blanket held straight by a wooden rod is released after the target has been covered. The third figure shows the resulting placement - the offset is due to a combination of GNSS drift during the placement maneuver and the blanket sliding off after being placed, due to the air flow from the fire analogue.

axes of its square platform and $1\,\mathrm{m}$ above the ground. Initially, all four of these positions are checked against a predefined safety area in order to prevent collisions with walls or the safety net. Of the starting positions that pass this check, the closest one to the current position of the UAV is selected.

The UAV flies to the position, orienting itself s.t. the UAV faces the estimated target. Since no measurement of the target is possible at this point due to the low altitude, the maneuver is executed without feedback based on the previous target pose estimate. The UAV then flies towards the target at the constant speed of $1\,\mathrm{m\,s^{-1}}$. When it reaches the distance of $3\,\mathrm{m}$ from the target, the blanket is unrolled as shown in Fig. 9. This is done by releasing the first of the two servo-controlled claws in the blanket holder on its underside and letting the blanket fall free. As it flies on, the blanket is fully unrolled by friction with the ground. The UAV then seamlessly spreads the blanket over the target as it passes over its estimated center. Once it reaches a point $0.6\,\mathrm{m}$ beyond the center - half the length of the blanket - the second claw is released, dropping the rod attaching the blanket to the UAV and thus concluding the placement. Afterwards, the UAV ascends to the safer height of $2\,\mathrm{m}$, and is set to return to the takeoff point for checkup, replacement of the blanket, and potentially also for replacement of the battery. In real firefighting, *e.g.*, for extinguishing a burning vehicle, the maneuver would be the same, with the only difference being that the UAV would have to be upscaled to be able to carry a sufficiently large blanket - *e.g.*, the fire blanket used in Fig. 2 with the size $6\times8\,\mathrm{m}$ - in order to sufficiently cover the entire target and block access to oxygen to the fire.

VIII. RESULTS

Before the competition, our team was performing numerous tests alternating between simulation and real world deployment. Simulation is faster and cheaper to prepare, but contains unaccounted-for differences due to simplifications when compared to the real world. Real world testing allowed us to address these, as well as to incrementally adjust the simulation to better correspond to observed reality. For simulations, we used the Gazebo simulator - a part of the Robot Operating System (ROS) package. In this simulator, we built the fire analogues used in the competition based on the provided specifications.

All the software developed for the competition, including the Gazebo plugins, is open-source and is available on-line[2].

In simulation, we could obtain a rough evaluation of the precision of the fire analogue localization precision thanks to the availability of the exact ground truth positions . Out of 10 deployments, the average position error was $0.139\,\mathrm{m}$ and the average orientation error was $0.125\,\mathrm{rad}$ (7.2°). The worst position error was only $0.175\,\mathrm{m}$ and the worst orientation error was $0.275\,\mathrm{rad}$ (15.8°), showing that the target localization method is sufficient for deployment.

Real world tests of this system were done in a desert location in the United Arab Emirates where we were based for six weeks prior to the competition. Our stay there was motivated by practical considerations - in our long-term experience, lighting and other local environmental conditions significantly affect real-world robotic deployment, particularly the sensor outputs. Therefore, preparing a robotic system for reliable operation in a certain location and climate requires testing in equivalent conditions. For instance, the stronger sunlight and higher temperatures in UAE affected thermal imaging in ways that were not apparent in our central European country.

In the UAE, we tested the blanket deployment against the local wind conditions as well as thermal detection.

The most significant testing results were shown in the competition challenge itself. Each of the three challenges progressed as follows: First, each participating team was given three rehearsal runs at $15\,\mathrm{min}$ each. Afterwards, two graded attempts were taken as part of the competition proper, $15\,\mathrm{min}$ each, and the final team result was selected as the better result of the two attempts.

In the competition, we were the one of only two teams to place a fire blanket on the fire analogue fully autonomously without RTK penalization (see Fig. 9 and videos[3]). In addition to the points achieved for this subtask of MBZIRC 2020, we demonstrated a fully functional system of the same complexity as would be required for real firefighting missions.

The most successful opposing team, *University of Seville, IST Lisbon and CATEC*, used a solution based on dropping the blankets from above with an attached rigging that spread the blankets upon impact with the target akin to a parasol. This solution was more efficient in achieving maximum possible overlap with the surface of the fire analogues compared to our dynamic spreading that induces drift w.r.t. the initial estimated target position. This approach benefited from releasing the blanket immediately after reaching a position above the target in combination with rapid dropping of the blanket in its folded form. However, it also shows less potential for subsequent real world firefighting, since the deployed rigging keeps the blankets taut in a horizontal plane as opposed to letting them fall over the target object to block oxygenation of the fire. The weight of the rigging further challenges the carrying capacity of Multi-

---

[2]`https://github.com/ctu-mrs`
[3]`http://mrs.felk.cvut.cz/mbzirc-2020-blanket-placement`

rotor UAVs, making it necessary to use larger units than would otherwise be the case. Lastly, the deployment method necessitates hovering for an extended period of time above a fire in its convection column, endangering the UAV itself. It should also be noted that this solution used RTK-GNSS for localization, with the positions of the fire analogues being pre-measured as opposed to located with onboard sensing - an approach that is not applicable for real firefighting and was therefore penalized in the competition as a solution with lower technology readiness level.

Another team that managed to place a blanket on the target, *Polytechnic University of Madrid, University Pablo Olvide, Poznan University of Technology, CNRS*, did so without using RTK-GNSS. Their target localization technique was based on thermal imaging similar to ours, using significantly heavier UAV platform. However, their method of blanket placement constituted dropping it directly from above the target, without any visible method designed to unroll and spread it out it first [20]. This is sub-optimal, as part of the limited size of the blanket is very likely to remain folded without extinguishing the fire.

Without an RTK base station, the precision of GNSS localization depends on the quality of the signal from satellites. GNSS satellites broadcast their signals from space, but what is received depends on additional factors including, for instance, signal blockage and its reflection and atmospheric conditions. For this reason, we have delegated the sensing necessary for obstacle avoidance on 2D LIDAR data to employ a virtual bumper that did not allow flight closer to obstacles than $5\,\mathrm{m}$.

Out of the 5 attempted blanket placements by our team in the entire competition, the average offset of the placed blankets from the center of the selected fire analogue was $\sim 1.0\,\mathrm{m}$. Given how the video documentation shows that the UAV hovered correctly above the selected fire analogue in the estimation phase, we can presume that the precision of the estimation was comparable to simulation. Considering these observed effects, it is desirable to have constant visual feedback during the placement maneuver for correction. Our system relying solely on thermal cameras was not suited for this with the fire analogues used in the competition, but in case of a real firefighting, a front-facing thermal camera would be sufficient to correct the drifting course of the UAV towards an object on fire.

IX. Conclusion

In this paper, an approach to extinguishing ground fires with fire blankets deployed by autonomous Multi-rotor UAVs was proposed. The system comprised an advanced autonomous UAV control, localization based on sensory fusion from multiple sources, thermovision and its specialized processing, target estimation, and a purpose-built blanket deployment hardware subsystem. This approach was shown as effective in the MBZIRC competition, but it also appears to be practical for real firefighting scenarios, in many cases over-performing classical concepts of spraying liquid fire extinguishing agents limited by the payload capacity of Multi-rotor UAVs. For the MBZIRC competition, this approach and system contributed to winning the overall first place in the Grand Challenge.

References

[1] L. Merino, F. Caballero *et al.*, "An Unmanned Aircraft System for Automatic Forest Fire Monitoring and Measurement," *Journal of Intelligent and Robotic Systems*, vol. 65, pp. 533–548, Jan. 2012.

[2] A. Restas, "Forest Fire Management Supporting by UAV Based Air Reconnaissance Results of Szendro Fire Department, Hungary," in *First International Symposium on Environment Identities and Mediterranean Area*, 2006, pp. 73–77.

[3] A. Viguria, I. Maza, and A. Ollero, "Distributed Service-Based Cooperation in Aerial/Ground Robot Teams Applied to Fire Detection and Extinguishing Missions," *Advanced Robotics*, vol. 24, no. 1-2, pp. 1–23, Jan. 2010.

[4] R. Chen, H. Cao, H. Cheng, and J. Xie, "Study on Urban Emergency Firefighting Flying Robots Based on UAV," in *IAEAC*, 2019.

[5] D. A. Saikin, T. Baca, M. Gurtner, and M. Saska, "Wildfire fighting by unmanned aerial system exploiting its time-varying mass," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2674–2681, 2020.

[6] H. Qin, J. Q. Cui *et al.*, "Design and implementation of an unmanned aerial vehicle for autonomous firefighting missions," in *ICCA*, 2016.

[7] P. Pecho, P. Magdolenová, and M. Bugaj, "Unmanned aerial vehicle technology in the process of early fire localization of buildings," *Transportation Research Procedia*, vol. 40, pp. 461–468, Jan. 2019.

[8] A. Imdoukh, A. Shaker *et al.*, "Semi-autonomous indoor firefighting UAV," in *ICAR*, 2017.

[9] H. Lee, H. Kim, W. Kim, and H. J. Kim, "An integrated framework for cooperative aerial manipulators in unknown environments," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2307–2314, 2018.

[10] A. Caballero, A. Suarez *et al.*, "First experimental results on motion planning for transportation in aerial long-reach manipulators with two arms," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 8471–8477.

[11] C. C. Kessens, J. Thomas, J. P. Desai, and V. Kumar, "Versatile aerial grasping using self-sealing suction," in *ICRA*, 2016, pp. 3249–3254.

[12] G. Loianno, V. Spurny *et al.*, "Localization, grasping, and transportation of magnetic objects by a team of mavs in challenging desert like environments," *IEEE Robotics and Automation Letters*, 2018.

[13] V. Spurný, T. Báča *et al.*, "Cooperative autonomous search, grasping, and delivering in a treasure hunt scenario by a team of unmanned aerial vehicles," *Journal of Field Robotics*, vol. 36, no. 1, pp. 125–148, 2019.

[14] T. Baca, P. Stepan *et al.*, "Autonomous Landing on a Moving Vehicle with an Unmanned Aerial Vehicle," *Journal of Field Robotics*, vol. 36, pp. 874–891, 2019.

[15] T. Baca, M. Petrlik *et al.*, "The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles," 2020, arXiv: 2008.08050.

[16] *User Manual for TeraRanger Evo Thermal with: USB and UART backboards*, Terabee, 2018.

[17] W. Minkina and S. Dudzik, *Infrared thermography: errors and uncertainties*. John Wiley & Sons, 2009.

[18] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258 – 1276, 2013.

[19] H. Choset and P. Pignon, *Coverage Path Planning: The Boustrophedon Cellular Decomposition*. London: Springer, 1998, pp. 203–209.

[20] S. Martinez-Rozas, R. Rey *et al.*, "Skyeye team at MBZIRC 2020: A team of aerial and ground robots for gps-denied autonomous fire extinguishing in an urban building scenario," 2021. [Online]. Available: https://arxiv.org/abs/2104.01834

Chapter 4

Research motivated by the *UVDAR* system

The development of the *UVDAR* system enabled the performance of a wide range of complex multi-UAV operations in real-world conditions. Such operations were previously either only theoretical, or were tested in non-realistic simulations or strictly controlled conditions [15], often using explicit global localization system in the loop as a substitute for true distributed relative localization systems [40]–[43].

This localization can only be provided by an external infrastructure, severely limiting the scope of applications for multi-UAV systems.

Deploying multi-UAV flights with *UVDAR* revealed several practical limitations of such systems not previously considered, making us aware of specific properties of the system that needed to be taken into account with novel algorithms. Additionally, beneficial properties of the system not previously expected organically arose, allowing us to enact novel flight behaviors and to solve tangentially related robotic challenges.

Swarm flight

A very often discussed modality of multi-UAV deployment is the *swarm*, where each unit - in literature referred to as an *agent* or *particle* - follows the same simplistic set of rules that forces it to maintain a certain relative distance or orientation w.r.t. its neighbors in a strictly decentralized manner. This kind of behavior has various emergent phenomena of interest to both robotics and biology communities[44], [45]. Among other beneficial properties, a swarm is inherently robust to failure of individual units and the computational requirements for its implementation are typically lower than for more explicit cooperation approaches. Due to these properties, swarm flights were investigated extensively in the robotics community. This modality was chosen as a baseline benchmark for establishing the suitability of *UVDAR* in real-world UAV cooperation.

To the best of our knowledge, we were the first group to have successfully enacted a multi-UAV swarm flight operating exclusively using the onboard sensors and processing of the agents without any dependence on communication or external infrastructure - a feat made possible with *UVDAR*. This flight was more challenging than expected by the previous theoretical research, since the precision of a real-world relative localization sensor is lower and less isotropic than in simulation models. This work is shown in the attached paper

The swarm deployment had numerous follow-up research endeavors, where we not only refined the quality of the sensor processing, improved the flight behavior reliability, and scaled up the swarm size [27], but we also reached another unique achievement - we were the first group to fly a swarm of UAVs within the cluttered, collision-prone environment of a forest

without any communication required for group stabilization [26]. In another work [25], we showed how temporarily changing the blinking signal allows a swarm member to use *UVDAR* to notify other members of the presence of an intruder, and thus to propagate a change in their behavior. The property of *UVDAR* that it inherently transmits digital data as identifiers was further exploited in [46], where we used the system for explicit optical data communication. This method allows for low-bandwidth transmission that is entirely resistant to radio interference and can be used in specialized sensitive cooperative tasks.

Automatically annotated image dataset for ML-based vision

In recent years [22], there was an explosive growth of ML-based computer vision systems, particularly based on artificial neural networks. Of interest among these technologies is image recognition of objects, made suitable for deployment onboard UAVs by miniaturization of computational acceleration hardware. This technology therefore had the potential to be used for VRL of UAVs, and our group was among the first to implement ML for the real-world aerial interaction of UAVs [23]. Due to the limited onboard computational resources, the ML-vision system used by a UAV needs to be constrained to address the specific application. This was, in our case, the detection of a nearby UAV target. A significant challenge in ML-based VRL systems is that their reliability is strongly informed by the quality of the training image dataset used. These detectors work well if the deployment environment and the target objects of interest are visually close to their equivalent in the training dataset. In general, this was highly challenging to achieve - obtaining such an annotated dataset required manual annotation of thousands of images for each environment of interest.

To address this challenge, we used *UVDAR* to generate a practical, real-world dataset of RGB images with annotated positions of UAVs in various environments viewed from the perspective of other UAVs. Automated annotation of images captured in flight was not previously possible, since global localization and ego-pose estimation systems are not currently precise enough to correctly reproject global UAVs positions obtained by systems such as GNSS onto the camera image space. The main issues involved include the drifting of the relative position due to GNSS error and, even more significantly, the drifting and hysteresis of orientation estimates of UAVs. The advantage of *UVDAR* in this context is that the system can be attached to the same rigid body as the dataset-gathering RGB camera, removing the need for transforming multiple noisy body pose measurements into the camera frame. Both cameras are, in essence, relative bearing estimators located at nearly the same position, eliminating the localization error of the observer from the reprojection of a detected target onto the RGB camera. Additionally, the *UVDAR* markers only minimally alter the visual appearance of a target UAV. Their lights can be optically filtered out from visible color images, making the dataset suitable for subsequent detection of UAVs *without* the markers. The dataset made using this technique was published with the attached paper

where the techniques used in producing it are described in detail. The publication of such a dataset promotes the use of multi-UAV flight algorithms with various VRL systems, es-

pecially with popular ML technologies. The dataset was used by researchers for analysis of the performance of ML vision algorithms for appearance-based [47] and motion-based [48] detection of UAVs.

Cooperative flight using relative orientation estimate

Another interesting property of *UVDAR* that we have exploited is that the individual markers on a single UAV can not only be distinguished to obtain the approximate distance of the UAV from the camera, but they can also each carry a different signal ID. Thus, among onboard relative localization systems for UAVs, our system provides the unique ability to estimate the relative *orientation* of a target unit through the knowledge of side-specific IDs. We have implemented a variation of the classical leader-follower flight with the unusual modification that the leader is followed *directionally*, meaning that the follower always attempts to remain not only at a specific distance from the leader, but also on a specific side. Such a flight made it necessary to design a specialized flight behavior that respects a minimum safety distance constraint, in order for the follower not to attempt to reach its designated pose by colliding with the leader. This research is shown in the attached journal paper

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, Jul. 2019, ISSN: 2377-3766. DOI: 10.1109/LRA.2019.2901683.

This paper was also presented in the ICRA conference in 2019 and promoted the *UVDAR* system in the multi-robot community, as evidenced by it being my most cited paper.

Formation flight robust to sensory noise

An important stream of research in the multi-robot community concerns *formation flight*. This is a type of cooperative flight that, in comparison to a swarm, strives to maintain a specific shape for the UAV group. It is, however, more challenging than swarm flights due to the significantly stricter requirements on precision, both in the terms of sensing and of actuation. Research previously done in this field made significant progress in terms of theoretical flight control systems, but was lacking in not considering the properties of a real-world relative localization system, often assuming it to be perfectly precise or by assuming a shared knowledge of global poses of individual UAVs [32], [49]–[51]. Knowledge of the statistical properties of the *UVDAR* system output measurements allowed us to advance prior theoretical flight formation methodology to applicability in real world flights, which we have demonstrated in my latest paper

V. Walter, M. Vrba, D. B. Licea, and M. Saska, "Distributed formation-enforcing control for uavs robust to observation noise in relative pose measurements," 2023. arXiv: 2304.03057 [cs.RO] Under review in Transactions on Robotics,

currently under review. In this work, the theory of formation-enforcing control together with the known theoretical and practical properties of *UVDAR* were used in the design of a method for how the distributed formation-enforcing control (FEC) action derived from formation error

gradient descent [52] can be modified to take into account the noise of a relative localization system. The proposed method is based on modifying "set-points" of the control action, such that they enforce a specific limit on the statistical probability of *overshooting* the desired formation by the UAVs. Testing in both simulation and in real world deployment shows that this method outperforms naive gradient descent, thus showing great promise for future applications. Notably, the tests also show that if the noise is too large, the UAVs may altogether fail to achieve a formation without the proposed method. I plan to follow this research direction in the future as there are many potential improvements to be made to the proposed FEC, and there is also much space for analytical work to model and address the complex interaction of a FEC with noisy sensor inputs.

Bio-Inspired Compact Swarms of Unmanned Aerial Vehicles without Communication and External Localization

Pavel Petráček[1] , **Viktor Walter** , **Tomáš Báča** , **Martin Saska**

Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, 166 36, Prague 6, Czech Republic
[1]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`, `martin.saska@fel.cvut.cz`

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 (UVDAR) 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`.

Submitted to: *Bioinspir. Biomim.*

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 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 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.

3

Figure 1: A compact aerial swarm of 3 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.
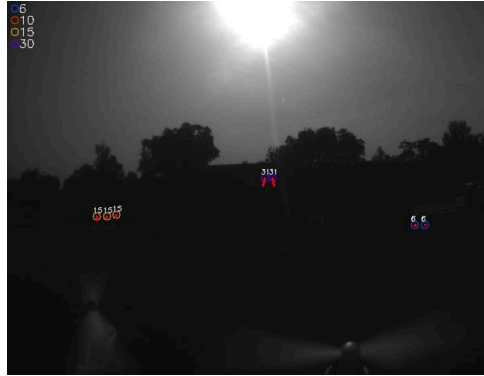
Figure 2: Onboard detection of 3 UAVs in the UV spectrum using UVDAR in a member of the aerial swarm. The method directly localizes the neighbors within a swarm in indoor and outdoor environments. Here, the method detects neighbors in an outdoor environment affected by a powerful source of ambient UV 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.

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 infra-structure-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 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 (CNNs). 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, 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 (SLAM) 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.

5

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	Infrared (IR) ranging
Nguyen et. al [6]	N/A	Yes	Ultra-Wideband ranging (UWB)
Nägeli et. al [26]	Yes	Yes	Visual markers
This work	Yes	**No**	UVDAR

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.

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:

(i) It provides an enabling technology for swarm research, often bio-inspired, by introducing a system that achieves fundamental swarm properties, as defined in [3].

(ii) 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.

(iii) 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.

(iv) It verifies the feasibility and analyses the usability of aerial flocking relying on direct localization, which is the most frequent mechanism in biological systems.

(v) It is based on several real-world deployments of aerial swarms.

(vi) It presents, to the best of our knowledge, the first autonomous deployments of aerial swarms with no centralized element and no mutual communication.

(vii) 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] and [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 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 3). 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 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 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

Figure 3: 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.

to ultraviolet 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 $30 \, \mathrm{J \, m^{-2}}$ weighted by the relative spectral effectiveness (unitless wavelength-specific factor). In our case of 395 nm radiation, this factor equals to 0.000036, making the actual limit $8.3 \times 10^5 \, \mathrm{J \, m^{-2}}$. 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 6 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 $\boldsymbol{v}_{[k]}^i \in \mathbb{R}^{3 \times 1}$, 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 $\boldsymbol{x}_{[k]}^{ij} \in \mathbb{R}^{3 \times 1}$ and relative velocity $\boldsymbol{v}_{[k]}^{ij} \in \mathbb{R}^{3 \times 1}$, $\forall j \in \{1, \dots, |\mathcal{N}_{[k]}^i|\}$, defined as

$$\boldsymbol{x}_{[k]}^{ij} = \left[x_{[k]}^{ij}, y_{[k]}^{ij}, z_{[k]}^{ij} \right]^T, \tag{1}$$

$$\boldsymbol{v}_{[k]}^{ij} = \frac{1}{\Delta t_{[k]}^{ij}} \left(\boldsymbol{x}_{[k]}^{ij} - \boldsymbol{x}_{[k-1]}^{ij} \right) - \boldsymbol{v}_{[k-1]}^i, \tag{2}$$

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 $\boldsymbol{v}_{[k=0]}^i = \boldsymbol{v}_{[k=0]}^{ij} = \boldsymbol{0}$. The swarming model is then defined as a sum of elementary forces

$$\boldsymbol{f}_{[k]}^i \left(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i \right) = \boldsymbol{f}_{[k]}^{b,i} \left(\mathcal{N}_{[k]}^i \right) + \boldsymbol{f}_{[k]}^{n,i} \left(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i \right), \tag{3}$$

where $\boldsymbol{f}_{[k]}^{b,i} (\cdot) \in \mathbb{R}^{3 \times 1}$ embodies the baseline forces as an interpretation of the *Boids model* [44] flocking rules *cohesion*, *alignment*, and *separation*, modified for real

UAVs as

$$\boldsymbol{f}_{[k]}^{b,i}\left(\mathcal{N}_{[k]}^i\right) = \frac{1}{\left|\mathcal{N}_{[k]}^i\right|} \sum_{j=1}^{\left|\mathcal{N}_{[k]}^i\right|} \left[\boldsymbol{x}_{[k]}^{ij} + \frac{\boldsymbol{v}_{[k]}^{ij}}{\lambda} - \kappa\left(\boldsymbol{x}_{[k]}^{ij}, R_n^i\right)\boldsymbol{x}_{[k]}^{ij}\right]. \tag{4}$$

The scalar λ [Hz] is the update rate of direct localization (camera rate) and the weighting function

$$\kappa(\boldsymbol{x}, r) = \max\left(0; \frac{\sqrt{\|\boldsymbol{x}\|_2}}{\|\boldsymbol{x}\|_2} - \frac{\sqrt{r}}{r}\right) \tag{5}$$

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 $\boldsymbol{f}_{[k]}^{n,i}(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i) \in \mathbb{R}^{3\times 1}$ 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 $\boldsymbol{v}_{[k]}^{n,i} \in \mathbb{R}^{3\times 1}$ 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

$$\boldsymbol{f}_{[k]}^{n,i}\left(\mathcal{O}_{[k]}^i\right) = \frac{1}{\left|\mathcal{O}_{[k]}^i\right|} \sum_{v=1}^{\left|\mathcal{O}_{[k]}^i\right|} \left[\frac{\boldsymbol{v}_{[k]}^{iv}}{\lambda} - \kappa\left(\boldsymbol{x}_{[k]}^{iv}, R_o^i\right)\boldsymbol{x}_{[k]}^{iv}\right] + \frac{\boldsymbol{v}_{[k]}^{n,i}}{\lambda}, \tag{6}$$

where the vectors of the relative position $\boldsymbol{x}_{[k]}^{iv} \in \mathbb{R}^{3\times 1}$ and the relative velocity $\boldsymbol{v}_{[k]}^{iv} \in \mathbb{R}^{3\times 1}$ 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

$$\boldsymbol{v}_{[k]}^i = \gamma\left(\boldsymbol{f}_{[k]}^i\left(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i\right)\right) \frac{\boldsymbol{f}_{[k]}^i\left(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i\right)}{\left\|\boldsymbol{f}_{[k]}^i\left(\mathcal{N}_{[k]}^i, \mathcal{O}_{[k]}^i\right)\right\|_2}, \tag{7}$$

where

$$\gamma\left(\boldsymbol{f}\right) = \min\left\{v_m; \lambda\|\boldsymbol{f}\|_2\right\} \tag{8}$$

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

$$\boldsymbol{r}_{[k]}^{d,i} = \frac{\boldsymbol{v}_{[k]}^i}{\lambda} \tag{9}$$

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 4. 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 $\boldsymbol{c}_{[k]}^{iv} \in \mathbb{R}^{3\times 1}$ referenced in the body frame of particle i, the state of a v-th virtual swarm particle is derived as

$$\boldsymbol{x}_{[k]}^{iv} = \left(1 - \frac{r_{[k]}^v}{\|\boldsymbol{c}_{[k]}^{iv}\|_2}\right)\boldsymbol{c}_{[k]}^{iv}, \tag{10}$$

$$\boldsymbol{v}_{[k]}^{iv} = \frac{r_{[k]}^v}{\|\boldsymbol{c}_{[k]}^{iv}\|_2}\left(\boldsymbol{I} - \boldsymbol{\mu}_{[k]}^{iv}\left(\boldsymbol{\mu}_{[k]}^{iv}\right)^T\right)\boldsymbol{v}_{[k]}^i, \tag{11}$$

where $\|\cdot\|_2$ is the L^2 norm, $\boldsymbol{I} \in \mathbb{R}^{3\times 3}$ is an identity matrix, and $\boldsymbol{\mu}^{iv} = \frac{\boldsymbol{c}^{iv}}{\|\boldsymbol{c}^{iv}\|_2}$. By analogy, the virtual swarm agent state can be derived for a linear obstacle defined by its normal vector $\boldsymbol{n}_{[k]}^{iv} \in \mathbb{R}^{3\times 1}$ and a set of observed points $\mathcal{P}_{[k]}^{iv}$ as

$$\boldsymbol{x}_{[k]}^{iv} = \left(\boldsymbol{I} - \boldsymbol{P}_{[k]}^{iv}\right)\hat{\boldsymbol{p}}_{[k]}^{iv} \tag{12}$$

$$\boldsymbol{v}_{[k]}^{iv} = \frac{1}{\|\hat{\boldsymbol{p}}_{[k]}^{iv}\|_2}\boldsymbol{P}_{[k]}^{iv}\boldsymbol{v}_{[k]}^i, \tag{13}$$

where

$$\boldsymbol{P}_{[k]}^{iv} = \boldsymbol{I} - \boldsymbol{n}_{[k]}^{iv}\left(\boldsymbol{n}_{[k]}^{iv}\right)^T, \tag{14}$$

$$\hat{\boldsymbol{p}}_{[k]}^{iv} = \arg\min_{\boldsymbol{p}\in\mathcal{P}_{[k]}^{iv}}\{\|\boldsymbol{p}\|_2\}. \tag{15}$$

CHAPTER 4. RESEARCH MOTIVATED BY THE *UVDAR* SYSTEM
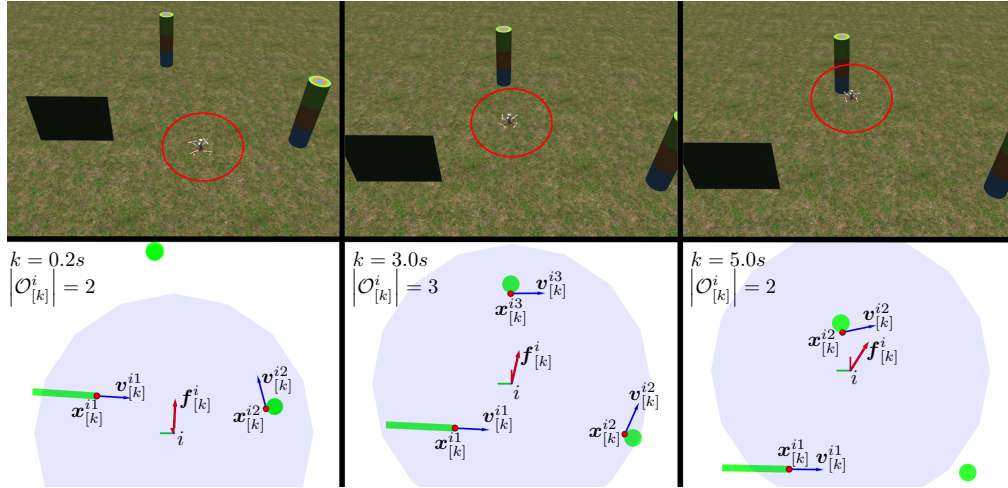
Figure 4: An autonomous UAV navigating among artificial obstacles according to the swarming model described in section 4. The UAV flies in the Gazebo robotic simulator (upper row), while it continuously detects geometrical obstacles represented as circles and lines in the onboard 2D laser-scanner data with a limited obstacle detection radius (gray circle). The states of virtual particles, consisting of position $\boldsymbol{x}_{[k]}^{iv}$ (red dots) and velocity $\boldsymbol{v}_{[k]}^{iv}$ (blue arrows) relative to UAV i, are visualized in the bottom image row. The steering force $\boldsymbol{f}_{[k]}^{i}$ (red arrow) of the swarming model represents the desired velocity.

The state of a virtual swarm particle for both geometrical classes is visualized in figure 4, 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 SO(3) state feedback controller [53], is depicted in the high-level scheme in figure 5. 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 5, the individual UAVs estimate their state vector

$$\boldsymbol{x} = [\boldsymbol{r}, \ \dot{\boldsymbol{r}}, \ \ddot{\boldsymbol{r}}, \ \boldsymbol{R}, \ \boldsymbol{\omega}]^T, \qquad (16)$$

where $\boldsymbol{R} \in \mathrm{SO}(3)$ is the attitude and $\boldsymbol{r} = [x_w, y_w, z_w]^T$ is the position in the world coordinate frame. The vector $\dot{\boldsymbol{r}} \in \mathbb{R}^{3\times1}$ is the linear velocity, $\ddot{\boldsymbol{r}} \in \mathbb{R}^{3\times1}$ is the linear acceleration, and $\boldsymbol{\omega} \in \mathbb{R}^{3\times1}$ 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

11

Figure 5: 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 $\mathbf{r}_{[k]}$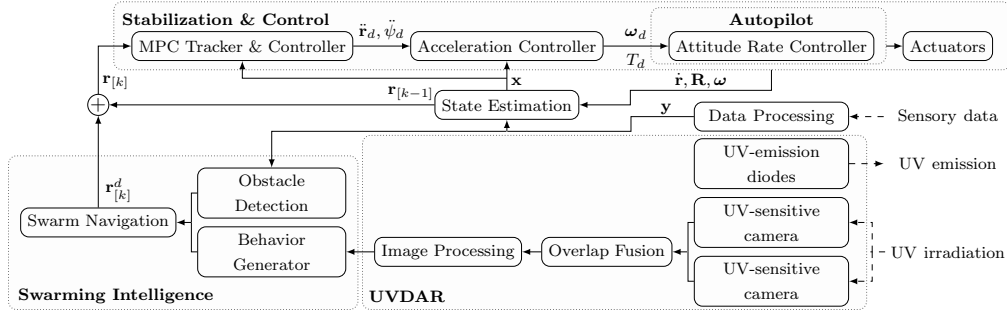 for the MPC in the MPC tracker, which outputs a command $\ddot{\mathbf{r}}_d$, $\ddot{\psi}_d$ (ψ is the heading acceleration) for the acceleration tracking SO(3) controller [53]. The acceleration controller produces the desired angular rate $\boldsymbol{\omega}_d$ and thrust reference T_d for the embedded attitude rate controller. A state estimation pipeline outputs the current state estimate $\boldsymbol{x}$ based on the sensory data $\boldsymbol{y}$ and the onboard measurements of linear velocity $\dot{\boldsymbol{r}}$, angular rate $\boldsymbol{\omega}$, and attitude $\boldsymbol{R}$. 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.

acceleration $\ddot{\boldsymbol{r}}$, the attitude $\boldsymbol{R}$, and the angular rate $\boldsymbol{\omega}$, using a combination of accelerometers, gyroscopes, and magnetometers. The embedded autopilot integrates the measurements of $\ddot{\boldsymbol{r}}$ to $\dot{\boldsymbol{r}}$ and employs the Extended Kalman Filter (EKF) 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 Spectrum ‡.

‡ `https://spectrum.ieee.org/automaton/robotics/drones/`
`video-friday-dji-mavic-mini-palm-sized-foldable-drone`

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.

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 6. 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 apriori available to the UAVs.

First, the collective dynamics of the swarming model are analyzed thoroughly in the Gazebo robotic simulator [57], shown in figure 4, coupled with the Robot Operating System (ROS) [58]. This simulation

Figure 6: Two distinct multi-rotor (hexa- and quad-rotor) UAV platforms, here equipped with UV-sensitive cameras (red) and with active UV markers (green), comprising the hardware components of the UVDAR system for relative localization of neighboring UAVs. The diagonal dimension (without propellers) of the platforms are 550 mm (left) and 450 mm (right). The hexa-rotor platform was used throughout our experimental verification presented in section 6.

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 7) 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 9 homogeneous units in a dense 3D forest-like environment, according to figure 8.

Second, an aerial swarm of 3 UAVs was experimentally deployed in a real-world forest-like environment similar to figure 8, 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 9, 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
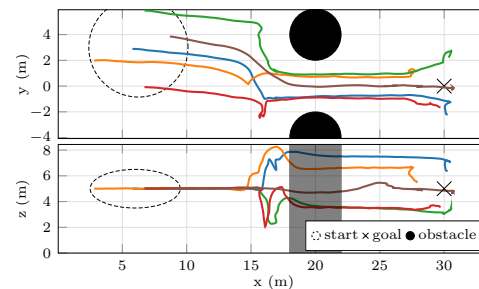
(a) Flexible and effective navigation of a decentralized swarm of 5 UAVs through a 4 m wide narrow passage.
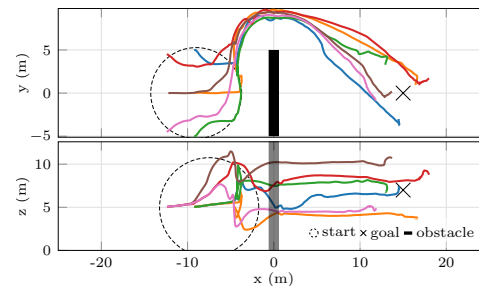
(b) Fast and efficient maneuvering of 6 UAVs emerging solely from local interactions during avoidance of a static obstacle.

Figure 7: A fully-decentralized swarm of homogeneous units in a simulated 3D environment with static obstacles. The swarming model yields enough flexibility for the compact team to deviate from its aggregated structure in order to pass safely through a narrow gap (a) or to avoid an obstacle in an efficient and fast manner (b).

(a) Orthogonal views on the trajectories of the particles.

(b) Average distance amid the UAVs.

(c) Minimal distance amid the UAVs.

(d) Minimal distance to the closest obstacle.
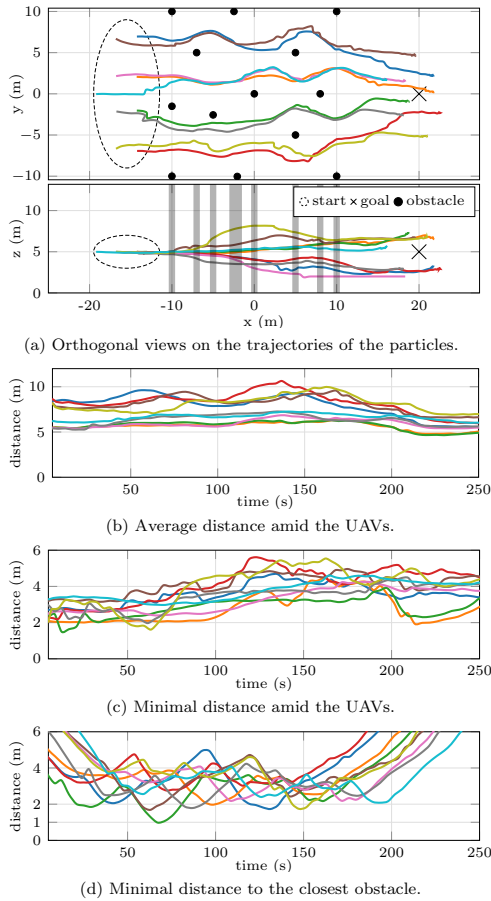
Figure 8: Navigation of a decentralized swarm of 9 homogeneous UAVs in a forest-like environment with a high density of circular obstacles – tree trunks (a). The experiment showcases the cooperative steering within the environment and the emerging properties of mutual long-term cohesion (b), safe mutual separation (c), and reliable obstacle avoidance (d).

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

(a) Swarm of 3 UAVs navigating through an artificial forest.

(b) Onboard RGB view from one of the homogeneous units.
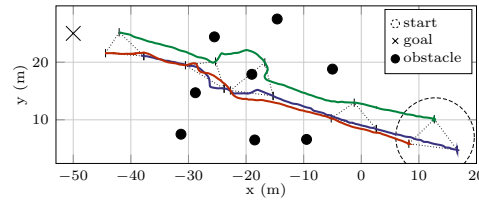
(c) Trajectories of individual UAVs (coded by color). The dotted triangles represent the swarm constellation (α-lattices) at a given time, which highlights the compactness and the flexibility of the swarm navigating amidst obstacles.
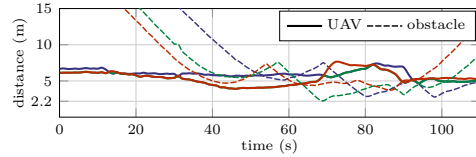
(d) Euclidean distance to the nearest UAV and obstacle for each swarm agent (coded by color). The minimum distance reached is 2.18 m.

Figure 9: Aerial swarm of 3 homogeneous UAVs in a real-world forest-like environment filled with artificial obstacles.

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 5, 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 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

15

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 10, 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 (figure 2 and figure 3 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
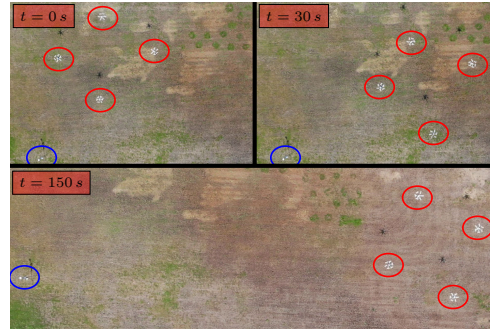
(a) Aerial view on the decentralized swarm of 4 UAVs (red) and a static reference to assist with the scale perception (blue).
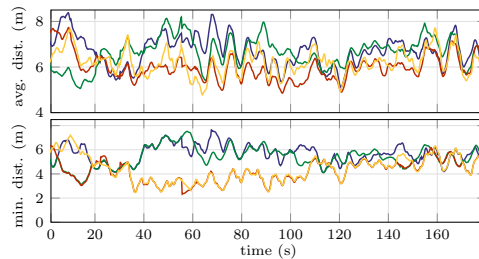
(b) Average and minimal Euclidean distance among the homogeneous agents. The minimum distance reached is 2.04 m.

Figure 10: A fully-decentralized swarm of 4 homogeneous UAVs navigating through an obstacle-less environment with UVDAR integrated into the position control feedback, as outlined by the scheme in figure 5.

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
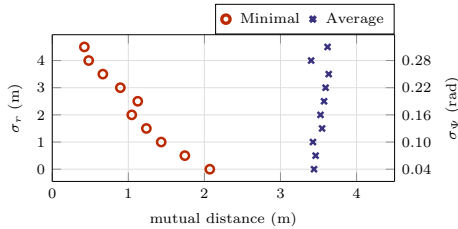
Figure 11: Dependency of the direct localization accuracy on the stability of an aerial swarm. The plot shows exponential decline of the minimum distance amid the swarm units with a growing degree of the localization error. The localization error is modeled as a multivariate normal distribution with uncorrelated zero-mean variables: the radial distance (standard deviation σ_r) and the relative azimuth (standard deviation σ_Ψ).

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 4 UAV units in an open environment (see figure 10) 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
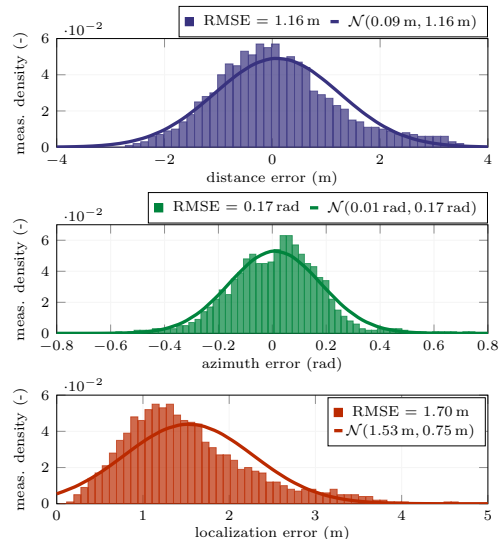
Figure 12: Quantitative accuracy of UVDAR direct localization with respect to GNSS positioning in real-world conditions. The figure shows error histograms and their normalized normal distribution $\mathcal{N}(\mu, \sigma)$ approximations of the directly estimated relative distance, the relative azimuth, and the global 3D position.

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
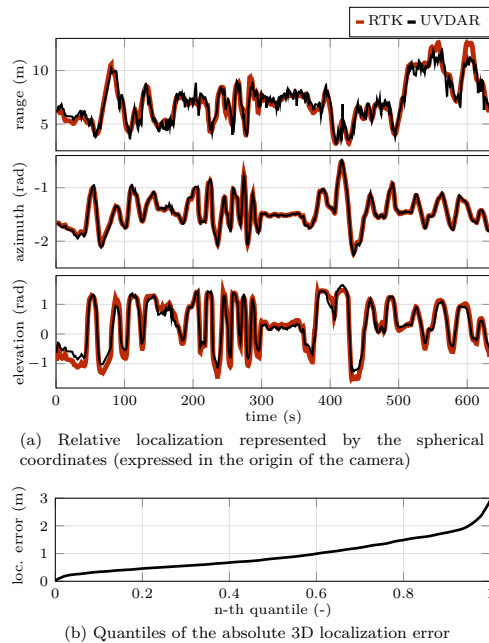
(a) Relative localization represented by the spherical coordinates (expressed in the origin of the camera)

(b) Quantiles of the absolute 3D localization error

Figure 13: Real-world accuracy of UVDAR direct localization in a controlled environment — tracking of a single mid-flight UAV relative to a static ground UV camera. The UVDAR localization is compared to ground-truth data obtained with the use of RTK-GNSS. The absolute RMSE of the relative 3D localization in this experiment reached 1.11 m (the median is 0.81 m).

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

Supplementary Materials

The multimedia materials supporting the article are available at `http://mrs.felk.cvut.cz/research/swarm-robotics`. The entire system is also available as open source at `https://github.com/ctu-mrs`.

ORCID

Pavel Petráček (iD): 0000-0002-0887-9430
Viktor Walter (iD): 0000-0001-8693-6261
Tomáš Báča (iD): 0000-0001-9649-8277
Martin Saska (iD): 0000-0001-7106-3816

References

[1] Štibinger P, Báča 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** 3634–3641

[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 5196–5202

[3] Trianni V 2008 *Evolutionary Swarm Robotics: Evolving Self-Organising Behaviours in Groups of Autonomous Robots (Studies in Computational Intelligence)* 1st ed (Springer Publishing Company, Incorporated)

[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 Computational Biology* **9** 1–7

[5] Chung S, Paranjape A A, Dames P, Shen S and Kumar V 2018 A Survey on Aerial Swarm Robotics *IEEE Trans. Robot.* **34** 837–855

[6] Nguyen T, 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** 3641–3648

[7] Bhavana T, Nithya M and Rajesh M 2017 Leader-follower co-ordination of multiple robots with obstacle avoidance *SmartTechCon*

[8] Saska M, Báča T, Thomas J, Chudoba J, Přeučil 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** 919–944

[9] Faigl J, Krajník T, Chudoba J, Přeučil L and Saska M 2013 Low-cost embedded system for relative localization in robotic swarms *Int. Conf. on Robotics and Automation* pp 993–998

[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** 1716–1726

[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 6885–6890

[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 891–898

[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 321–324

[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 6837–6842

[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** 30480–30490

[16] Intel® 2019 Drones Light Up The Sky URL 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 *Bioinspir. 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 3866–3873

[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**

[20] EHang 2019 EHang Drone Formation Flight URL 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 5015–5020

[22] Bürkle A, Segor F and Kollmann M 2011 Towards Autonomous Micro UAV Swarms *J. Intell. Robot. Syst.* **61** 339–353

[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 Trans. Robot. Autom.* **3** 1801–1807

[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* 4641–4647

[26] Nägeli 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 1141–1146

[27] van Diggelen F and Enge P K 2015 The World's first GPS MOOC and Worldwide Laboratory using Smartphones *ION GNSS+* pp 361–369

[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** 358–371

[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** 537–570

[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 155–160

[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

[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 819–824

[33] Shah D and Vachhani L 2019 Swarm Aggregation Without Communication and Global Positioning *IEEE Trans. Robot. Autom.* **4** 886–893

[34] Olaronke, Iroju and Ikono, Rhoda and Gambo, Ishaya and Ojerinde, Oluwaseun 2020 A Systematic Review of Swarm Robots *Current Journal of Applied Science and Technology* **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 *Rob. Auton. 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 Přeučil L 2014 Swarms of Micro Aerial Vehicles Stabilized Under a Visual Relative Localization *Int. Conf. on Robotics and Automation* pp 3570–3575

[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 Trans. Robot. Autom.* **4** 2637–2644

[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 Physics* **87** 171–186

[42] ProLight Opto Technology Corporation 2013 ProLight PM2B-1LLE 1W UV Power LED Technical Datasheet

[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 Ann. 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** 401–420

[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* 9080–9086

[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] Curiac 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 *Ph.D. 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 555–559

[52] Báča T, Petrlík M, Vrba M, Spurný V, Pěnička R, Heřt 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 5420–5425

[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 Trans. Robot. Autom.* **5** 2078–2085

[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 6753–6760

[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. Robots* **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 2149–2154

[58] Stanford Artificial Intelligence Laboratory et al Robot Operating System URL `ros.org`

On training datasets for machine learning-based visual relative localization of micro-scale UAVs

Viktor Walter*, Matouš Vrba* and Martin Saska*

Abstract— By leveraging our relative Micro-scale Unmanned Aerial Vehicle localization sensor *UVDAR*, we generated an automatically annotated dataset MIDGARD, which the community is invited to use for training and testing their machine learning systems for the detection and localization of Micro-scale Unmanned Aerial Vehicles (MAVs) by other MAVs. Furthermore, we provide our system as a mechanism for rapidly generating custom annotated datasets specifically tailored for the needs of a given application. The recent literature is rich in applications of machine learning methods in automation and robotics. One particular subset of these methods is visual object detection and localization, using means such as Convolutional Neural Networks, which nowadays enable objects to be detected and classified with previously inconceivable precision and reliability. Most of these applications, however, rely on a carefully crafted training dataset of annotated camera footage. These must contain the objects of interest in environments similar to those where the detector is expected to operate. Notably, the positions of the objects must be provided in annotations. For non-laboratory settings, the construction of such datasets requires many man-hours of manual annotation, which is especially the case for use onboard Micro-scale Unmanned Aerial Vehicles. In this paper, we are providing for the community a practical alternative to that kind of approach.

I. INTRODUCTION

The growing amount of processing resources sufficiently portable for deployment onboard lightweight MAVs has made it possible to run machine learning-based image processing on these devices in real time. This development is a crucial step towards visual relative localization of unmarked MAVs by other MAVs. This kind of localization is primarily useful for two applications: first, for detecting of cooperating MAVs in a swarm, a formation or otherwise operating friendly units, without the need to equip them with explicit markers or transmitters.

A second, more significant application is for detecting non-cooperating units, where marking them is not possible. These applications include, but are not limited to, reporting or eliminating MAVs intruding into a protected area, avoiding collisions in areas with unrelated active MAVs, or establishing cooperation with foreign units that the observer MAV encounters, if these are open to such a rapport.

For these reasons, markerless detection and relative localization of nearby flying aerial vehicles are topics that have been recently attracting interest of the robotic community

*Faculty of Electrical Engineering, CTU in Prague, Technická 2, Prague 6, {viktor.walter|matous.vrba}@fel.cvut.cz martin.saska@fel.cvut.cz

Fig. 1: Examples of annotated footage from the proposed dataset from a disparity of environments. The bounding boxes were generated automatically, using our relative localization system *UVDAR*.

[1]–[8]. Deep-learning based detectors have surpassed traditional detection methods in detection precision and robustness in general detection problems [9]–[11]. However, not many researchers are working on the use of deep-learning methods for the detection and relative localization of MAV. MAVs can be used in large quantities due to their low cost and their greater safeness than large UAVs. A large, varied and labeled dataset is a prerequisite for using any deep-learning based methods to train the classifier or regressor. These datasets are usually meticulously labeled manually, which is an arduous task. In this paper, we will address what we see as a significant reason why deep-learning is not used more often for MAV detection and relative localization, *i.e.*, the lack of suitable datasets and the lack of a simple and automatic way to generate such datasets.

In addition to machine learning (ML)-based vision, relative localization can be retrieved from absolute positions, obtained by a Real-time kinematic (RTK) global navigation satellite system (GNSS) [12] or by Motion capture (mo-cap) systems [13]. We deem this approach entirely unsuitable for field deployment, since both systems require lengthy setup and direct access of the operators to the operational space, which limits the size of this space. Other methods include measuring the relative strength of a radio signal, such as in [14], which however requires multiple observers or specific motion [15] to retrieve the full relative position. In addition, these systems are susceptible to interference.

Another relevant technology is LIDAR, which captures the surrounding surfaces as angularly distributed sample points, represented as a point cloud. These points are obtained by rapidly reorienting laser range sensor or sensors in a rotary manner and combining their measurements with the known current orientation. Using such system for relative localization of MAVs is problematic due to the small size and thin structures of these targets compared to the typical angular density of LIDAR rays, as well as due to the need to distinguish the small clusters representing MAVs

4.2. ARTICLE 8: AUTOMATIC TRAINING DATASET GENERATION FOR ML-BASED VISION

from noise and background. Approaches that are better suited for full onboard operation are based on vision. If a stereoscopic system with a sufficiently wide baseline [1] is available, flying MAVs may be retrieved on the basis of their distance from the surroundings. These systems are unfortunately expensive, large and require a great amount of processing power. A simpler approach is to mark MAVs with easy-to-detect passive visual markers, as in [16], [17]. These are inexpensive and are easily manufactured, but for the typical mutual distances of flying MAVs they need to be impractically large and are susceptible to adverse lighting conditions, particularly bright outdoor sunlight and shadows.

Active markers can also be used, as in [18], or in our own system UltraViolet Direction And Ranging (*UVDAR*) specialized for use in large compact swarms of cooperating MAVs. In this project, *UVDAR* is used for bootstrapping ML systems (see below). The obvious disadvantage of using artificial markers is that these methods are unsuitable for the detection of unmarked units, in addition to the fact that they are additional equipment that needs to be applied. These drawbacks do not apply at all to ML.

We hold that it is imperative to mitigate the main drawback of ML vision - the need for complicated dataset building.

While multiple benchmarking datasets for ML vision are available [19]–[21], these are primarily aimed at evaluating various qualities of a given system, and to the best of our knowledge no dataset has been released with the explicit goal of real-world deployment of MAV that can localize other MAV units. In particular, no system for automatic generation of such a dataset is currently available. An automated approach for this task can significantly expand the usability of ML-based vision, where some source of the ground-truth relative positions of targets has to be used for pointing out their image positions.

With MAVs, it is very difficult to retrieve reliable and precise orientation measurements, that are needed for correct projection of their relative positions into the image space of an on-board camera. The main challenges in this are the combination of the hysteretic properties of magnetometric sensors and their susceptibility to metal in environments, and also the insufficient scale of MAVs for orientation measurements from two absolutely localized body points using GNSS. Since our task primarily requires good relative bearing estimates, other vision-based systems appear to be the best candidate for the source of ground-truth.

Large passive visual markers, however, alter the appearance of MAVs too much to be suitable for training ML systems for detecting unmarked units, as the system would tend to specialize in detection of these markers, which are not expected to appear in final deployment. To address this challenge, we have developed an innovative system that allows for automated generation of annotated datasets for training ML systems that can be deployed for detecting MAVs from other MAVs. The system consists of a color camera attached to an observer MAV, as the source of the images for ML training, and also our specialized sensor for mutual relative localization *UVDAR* (see section II),

Fig. 2: Experimental platform views, with a passive visual marker (left) [22] and active ultraviolet markers, on and off, (center – left). Without the passive marker, the resemblance to arbitrary MAV is higher, making it more suitable for use in training ML algorithms.

an onboard computer and blinking ultraviolet LED markers attached to one or more target MAVs. Thanks to their small size and the fact that they radiate predominantly outside the visible wavelengths, these markers have minimal effect the visual appearance of the target (see Fig. 2, 3). This makes it possible to apply the trained ML system subsequently for detecting similar MAVs that do not carry these markers. These elements allow for easy and fast detection, localization and subsequent annotation of the areas in the camera images where the target MAVs can be found.

A unique feature of this system is that the above is done while circumventing requirements such as communication - liable for interference, blocking, congestion and other issues - or any source of absolute localization, such as RTK GNSS or a mo-cap setup. Our system addresses these limitations by applying relative measurements from one camera-based sensor into another, thus exploiting their known mutual orientation and distance, and also the difference in the wavelength ranges. Additionally, the presented system does not rely on any pre-existing infrastructure (such as a base-station and satellite visibility in the case of RTK GNSS or expensive, pre-calibrated camera setup in the case of mo-cap) in the deployment area, enabling fast and easy creation of labeled datasets in new environments. Additionally, the targets only require small LED markers as opposed to large antennas of RTK systems, making this approach more suitable even for very small MAVs. Here, we provide a large open-ended dataset MAV Identification Dataset Generated Automatically in Real-world Deployment (MIDGARD) for use by the community, to enable various ML systems to be trained and tested. The datasets comprise sets of images from continuous color camera footage of MAVs in a wide range of environments and backgrounds, together with annotations in the form of bounding boxes containing the MAVs in question, as well as their approximate distances.

II. UVDAR

For relative localization of surrounding MAVs used as a ground truth for labeling pictures in ML datasets, we propose to apply our system called *UVDAR*, described in detail in [23]–[25]. The system is based on computer vision in the ultraviolet (UV) range of radiation. This exploits the observation that sunlight is significantly weaker in UV than in the visible spectrum, allowing for easy detection of active UV markers in an effectively arbitrary indoor or outdoor environment by removing most other data from the image with the use of simple optical filtering. *UVDAR* sees active UV LED markers attached to cooperating MAVs as
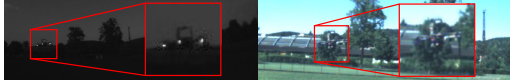
Fig. 3: Example of the view from the UV camera used in our *UVDAR*, system compared with a simultaneous view from the color camera. Note the apparent relative brightness of the markers in UV, while they are invisible in the color image. ML trained systems on such a dataset will be able to detect MAVs without markers.

small bright points. If multiple such points, belonging to the same MAV, are seen, geometrical considerations are used to estimate the distance of the vehicle from the sensor.

Since the markers would otherwise appear identical to each other, we enriched their information content by setting them to blink at a defined frequency, and we used a specialized algorithm to retrieve these frequencies as identifiers, in addition to retrieving the image positions of the markers even in the off phase of the blinking, when they would otherwise be invisible. With such information, we can either distinguish between individual MAVs, as is done in this paper, or we can retrieve their relative orientations by distinguishing different sides of the MAV with different frequencies [25].

The *UVDAR* system provides accurate bearing information on the neighbors, as well as estimates of their distances from the sensor. The sensor has a $180°$ field of view, with bearing errors of *approx.* $0.3°$, and a detection range of *approx.* $15\,\text{m}$, with typical error of 10-20 % of the target distance. Detailed experimental and analytical evaluation of the precision of relative position estimation by *UVDAR* is provided in [23]. Since the specific goal here is to annotate the image of another camera affixed to the body of the observer MAV carrying *UVDAR*, the precision of the bearing information is more significant. This is because a camera is essentially a device that converts the bearings of the points in its surroundings into pixel positions in its output image.

The targets are equipped with the UV LEDs, mounted on their exreme points, *i.e.*, typically the ends of their arms. These markers should have their output power set to account for the maximum expected distance from the observer. With our configuration, we use LEDs with a peak at $395\,\text{nm}$ and a Lambertian radiation pattern. We drive these at $170\,\text{mA}$ to produce $230\,\text{mW}$ of radiated power. Since these markers radiate predominantly in the near-UV wavelength outside the visible spectrum, they have limited influence on the image of a color camera, which typically separates its color channels with miniature color band-pass filters applied to its imaging elements. As stated above, these active markers have to blink with specified signals. There are three reasons for this: 1) The signals serve for identifying a specific marker, which enables multiple targets to be distinguished. 2) It makes the system more robust to specular reflections of the sun *e.g.*, from metallic corners, based on the observation that these do not blink as expected. 3) Specifically for this project, the blinking reduces the influence of the markers on the appearance of the targets. Depending on the exposure rate of the color camera, this is either due to apparently dimming them, or by producing frames where these markers are invisible because they were in the off-phase of the blinking.

III. Dataset generator

The aerial platforms that were used for generating the MIDGARD dataset also serve as an example of the equipment needed for other users to deploy the proposed system. Two types of MAV are involved - targets and observers. Targets are units serving as templates that the ML algorithms will train to detect. Observers are MAVs equipped with our special vision-based suite that generates image streams and annotates the positions of the targets within these images.

The observer units are equipped with two cameras attached to the same holder, the first being the *UVDAR* sensor (see section II) for relative position measurements, while the second is the camera producing the images for ML training. Any properly calibrated camera can be used, ideally of the same type as will be used for ML-based MAV detection.

In our datasets, we used *mvBlueFOX MLC200wC* camera sensors with a global shutter, with different lenses for different parts of the dataset. To calibrate them, we used the *OCamCalib* [26] omnidirectional calibration suite, since this toolbox accounts very well for lens distortions near the edges of the images, allowing for correct annotation across the whole image plane of the color camera. The two sensors are attached $8\,\text{cm}$ apart along the horizontal plane, perpendicular to both of their approximately parallel optical axes. Due to such compact installation, their mutual relative poses will only affect the projection of the measured positions into the image by their orientation component.

The processing of the *UVDAR* sensor data and also storage of the color camera stream and the performance of flight-essentials computations, are done with *Intel NUC* onboard computer. A short video demonstration of the system can be found at `mrs.felk.cvut.cz/midgard`.

A. Data acquisition

The UV camera is set to capture views at least at $70\,\text{FPS}$, to allow for the retrieval of a blinking signal of up to $30\,\text{Hz}$, below the Nyquist frequency. The raw input images do not need to be stored at this full frame-rate, as performing such rapid storage access operations tends to reduce the actual frame-rate of the detection system significantly. Instead, only detected active maker positions are stored in each frame.

The color camera footage is recorded by default at $3\,\text{Hz}$ to avoid producing numerous frames of high similarity, but it can be increased (*e.g.*, for applications that use inter-frame tracking). The subsequent processing steps are performed onboard of the observer in real time, so that the raw datasets are available immediately after the flight. They can also be done offline after deployment, if the observer does not possess sufficient computational resources, or if the operator wishes to adjust the settings of the consecutive processing steps, the outputs of which are all stored.

(a) Relative pose estimation from *UVDAR* input (see [25])

(b) Measurement covariance estimated with UT on the input

(c) Filtered pose estimate is transformed into the color camera frame

(d) The pose estimate is projected into the color camera using UT
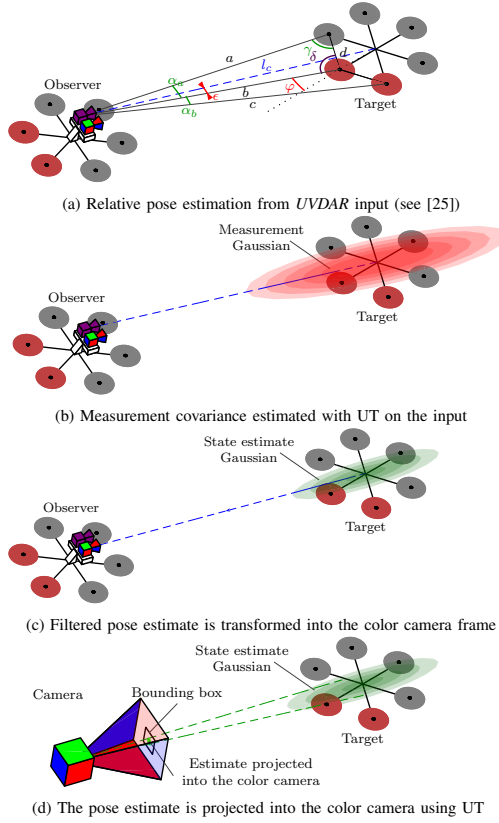
Fig. 4: Consecutive phases of dataset generation - automatic pose estimation and reprojection

B. Detection and localization

As described in [25], the annotated markers detected on a target MAV can be used to retrieve an estimate of its relative pose. In the proposed system, a new position estimation approach needed to be designed. Since the camera producing the dataset views is not synchronized, and is even potentially delayed w.r.t. the *UVDAR* sensor, it is necessary to use a tracking mechanism that allows for the retrieval of relative pose estimates for the instants when images are produced. A linear Kalman filter, which additionally provides robustness to the target being temporarily lost from view due to occlusion or exiting the field of view of the sensor, is used as a core mechanism in the proposed system. To be input into the Kalman filter, the relative measurements must include covariances, approximating the measurement noise or a measure of the reliability of a given measurement (Fig. 4b). The precision of the *UVDAR* measurement depends primarily on the ratio between the resolution of the UV camera, the field of view of its lens and the distance of the

target, due to perspective foreshortening.

The precision of a measurement in the form of a Kalman filter-compliant multivariate Gaussian distribution is difficult to express analytically, due to the highly non-linear dependence between the image positions of the markers and the 3D pose of the object carrying them. Unscented Transform (UT) [27] is used to translate the known precision of detection of the markers in the *UVDAR* image into the approximate covariance of the 3D pose of the MAV itself. The input to the transform is a vector containing the image positions of the markers belonging to a given MAV, mean values of the variables that express the ambiguities in some cases of the detection, and also an error covariance matrix for all of these variables. The values expressing the ambiguities are set to their mean expected values, *e.g.*, the angle by which the target MAV is rotated away from perpendicular alignment if only two markers are seen. The input covariance matrix thus expresses the error in the image positions stemming from pixel size and the image processing involved as well as the approximate ranges in the known ambiguities. For example, the input vectors for observing two and three adjacent markers on a hexarotor, as described in [25], are

$$\mathbf{x_2} = \begin{bmatrix} x_1, \, y_1, \, 1/f_1, \, x_2, \, y_2, \, 1/f_2, \, \delta=0, \, \alpha=0, \, \varphi=0 \end{bmatrix}^T \quad (1)$$

$$\mathbf{x_3} = \begin{bmatrix} x_1, \, y_1, \, 1/f_1, \, x_2, \, y_2, \, 1/f_2, \, x_3, \, y_3, \, 1/f_3, \, \beta=0 \end{bmatrix}^T . \quad (2)$$

Here, x_i, y_i and f_i are the measured image coordinates of each marker i and the measured frequency of its blinking, to account for the possibility of incorrect matching with a known template. Symbol δ refers to the angle by which two observed markers differ from the pose where their connecting line is perpendicular to the line of sight of the observer, while α represents the ambiguity in orientation when observing two markers of the same ID on a MAV with six markers of only two different marker IDs. Element φ represents the unknown amount of tilt that the target has w.r.t. the line of sight of the observer. Symbol β in three-marker observation, where the above ambiguities do not apply, resolves an observation of an ID sequence that does not fit the known marker layout, by introducing a wide additive orientation component in that case. The input measurement covariances entering into the UT for each situation are

$$\mathbf{P_2} = \text{diag}\left(\begin{bmatrix} e_x^2, e_x^2, e_p^2, e_x^2, e_x^2, e_p^2, e_\alpha^2, e_\delta^2, e_\varphi^2 \end{bmatrix}^T\right) \quad (3)$$

$$\mathbf{P_3} = \text{diag}\left(\begin{bmatrix} e_x^2, e_x^2, e_p^2, e_x^2, e_x^2, e_p^2, e_x^2, e_x^2, e_p^2, e_\beta^2 \end{bmatrix}^T\right) \quad (4)$$

where $\text{diag}(\mathbf{x})$ represents a diagonal matrix with elements of $\mathbf{x}$ on the main diagonal, e_x is the mean image position error in pixels, e_p is the mean error of the blinking period measurement, and the rest of the variables represent the mean ranges of the associated ambiguities. We set the values on the basis of geometrical assumptions, and we refined them empirically to the following values: $e_x = 2, e_p = 0.2/f_c, e_\alpha = \pi/20, e_\delta = \pi/3, e_\varphi = \pi/18, e_\beta = 2\pi/3$ where f_c is the current UV camera frame-rate.

The input measurement vectors $\mathbf{x_2}$, resp. $\mathbf{x_3}$, together with the appropriate input covariance, are used with the UT to

produce a set of sigma-points, representative of the spread of the values of the vectors. These sigma-points are each converted into a relative target pose estimate, as described in detail in [25], where some are affected by the ambiguity elements being non-zero. These output poses are combined into a single weighted average, and their spread is used to approximate the error covariance of the final pose estimate. For other marker layouts, we progress equivalently. Note that measuring a blinking frequency that is close to two expected values will increase the error covariance, since for certain sigma-points some of the observed markers can be matched with different body markers than others in the 3D pose calculation. The output covariances are invariably strongly elongated in the direction from the detector to the target, showing the characteristic property of visual localization that the distance estimate is significantly less precise than the bearing of the observed object. Since the detection of only two markers in the image contains more ambiguity than with three, the covariances are larger for the former.

If only a single marker is detected (due to an occlusion or of large distance of the target) no distance information is retrieved, except for the known detection range that provides the upper distance limit. The relative position of the marker can therefore be anywhere along its corresponding optical line, up to the maximum detection distance. The markers lie on the extreme points of the target, and the marker that is currently detected may, from the perspective of the observer, lie on the silhouette of the target. The center or the target MAV can therefore reasonably be expected to lie inside a cylinder, the longitudinal axis of which points towards the detected marker, with radius equal to the maximum distance of the markers from the target MAV center. For use in a Kalman filter, this cylindrical subspace is approximated by an elongated Gaussian. While this specific case of measurement is less informative than with multiple markers, it still proves useful with a Kalman filter if a better prior estimate initiated the filter with a distance estimate. In that case the change from the previous bearing can preserve reasonably precise tracking. Without applying this new information the process noise of the filter would expand the state covariance beyond useful size, in addition to the mean value not following the changing pose of the target at all. Furthermore, for the purposes of image annotation, the lacking distance information is admissible, since the *UVDAR* sensor and the color camera - both essentially bearing sensors - are close enough to each other for the reprojection of the covariance.

C. Data post-processing

The estimates of the relative poses of the target MAVs in the frame of the *UVDAR* camera are input into a linear Kalman filter, since the estimates are expressed in Cartesian 3D coordinates (Fig. 4c). Since the relative pose between the *UVDAR* camera and the color camera is fixed and known, the transformation into the camera frame from the external frame will negate the effects of inevitable errors in the absolute observer pose estimate. This is possible because both the transformation from the *UVDAR* frame to an external frame
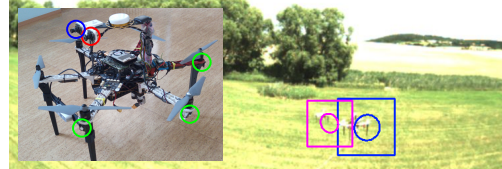
Fig. 5: The MAV platform used in our experiments, here equipped both with a *UVDAR* camera (red) and with a color camera (blue). The markers (green) do not need to be applied to the observer unit, and here they merely demonstrate their layout on the target. In the background, two target MAVs overlap, but are still both detected since some markers of each are seen by *UVDAR*.

and the transformation from an external frame to a color camera frame are burdened by the same error in opposite directions. Correction step of the Kalman filter is applied when a new UVDAR measurement is available. A state of the filter at the time of the measurement is predicted using the latest available filter state, and then the measurement is used to correct it. To obtain a state estimate at the current time, another prediction step is applied to the corrected state. This approach enables avoiding incorrect estimates due to camera delays.

The annotations provided by the proposed system with the camera footage take the form of bounding boxes enclosing the target MAVs in the image, in addition to the estimated range of target distances. The last step of converting the relative pose estimates into these bounding boxes is reprojection, which is done by applying the UT to the current relative position estimates, transformed into the color camera frame (Fig. 4c), in order to obtain a projection of the position estimates, including the covariances, into the image of the color camera (Fig. 4d). This 2D covariance is then converted into a rotated ellipse, by selecting a boundary probability level, *e.g.*, 2σ, but since computer vision ML systems typically [9]–[11] work with rectangular areas, an axis-aligned rectangular bounding box is derived from the ellipse, and is then further expanded.

During creating of MIDGARD, over 85% of the images, containing MAVs in range of the UVDAR system, were automatically labeled with sufficient precision to be included in the final dataset. This outcome is representative of the capabilities of the proposed system, as the values are similar in all recorded scenarios. The pictures where no UAV was detected (most of the incorrectly labelled pictures) were automatically deleted and some remaining outliers were removed using a GUI software that allows the user to discard ranges of annotated views that they consider unsuitable for a specific application.

IV. MIDGARD DATASET

A. Platforms

The experimental platforms used for generating the MIDGARD dataset are based on the *DJI F550* frames, equipped with *Pixhawk* flight controllers and *Intel NUC*

computers. In theory, for our method of dataset generation the computers are needed only for the observer units, while both the targets and the observers can even be piloted manually if no other option presents itself. For MAVs that are highly visually dissimilar to the model above, new footage has to be made. Since MIDGARD is an open-ended dataset, footage of other models will be provided in the future.

B. Environments

The dataset provided was gathered in indoor and outdoor environments. The outdoor parts of the dataset were obtained in various locations, including forest, meadow, fields and urban facilities. These showcase various backgrounds that can be found outdoors, including trees and fields as well as houses and repetitive man-made structures (see table I). The indoor footage includes complicated indoor backgrounds and lighting conditions. The trajectories used for the observer and target MAVs were designed in such a way that the dataset they produce will contain a number of significant challenging situations, in addition to normal views with MAVs against a full variety of backdrops in the given area at a full operational range of distances from the observer. These situations include temporary loss of line-of-sight by a target MAV leaving the field of view of the observer and UAV eclipsing each other in footage with two targets.

C. Examples of main locations

1) Countryside: The footage gathered in the countryside involves observation of two target MAVs, presented against various backdrops - fields, hills, deciduous trees, a distant village and a coniferous forest. Notably, in this footage the two targets eclipse each other in the view (Fig. 5). This is intentional, and was ensured by applying the Model predictive control (MPC) tracker with predefined trajectories.

2) Semi-urban landscape: This footage includes backgrounds of wide buildings, distant hills, vehicles and covered stands. The footage contains one MAV as a target.

3) Classical interiors: Currently contains footages from a vestibule in our departments with ornate stairwells, arched windows and stuccoed ceilings and a historical church under reconstruction, obtained under our DRONUMENT project[1].

These represent complex backgrounds that can be encountered inside and outside of historical buildings. Both footages have one MAV as a target. Notably, due to the low lighting in the vestibule, the exposure levels of the color camera were high enough for the markers to be visible, which was addressed in the church footage by applying low-pass optical filter onto the color camera.

4) Modern interiors: This footage represents a modern, utilitarian architectural background representative of what can often be seen inside and outside modern buildings, with one MAV as the target. The first setting here is a transitory room built inside former courtyard of a building belonging to the faculty of mechanical engineering of the Czech Technical University. The room has glass walls, framed by steel beams,

[1]mrs.felk.cvut.cz/research/historical-monuments-documentation

Background	Lighting	FoV	Frames
Fields, hills	Direct sunlight	180°	780
Fields, hills	Direct sunlight	96°	554
Coniferous forest	Direct sunlight	180°	763
Coniferous forest	Direct sunlight	96°	769
Semi-urban	Direct sunlight	96°	475
Stands	Direct sunlight	96°	586
Modern architecture	Strong indirect natural light	96°	534
Historical stairwell	Low light through windows	96°	319
Church interior	Very low mixed light	96°	984
Church exterior	Overcast, late evening sky	96°	697
Warehouse interior	Low fluorescent lightbulbs	96°	564
Warehouse exit	Changes halfway	96°	272
Appartment buildings	Overcast sky	96°	300

TABLE I: Primary characteristics of the current dataset

and an uncovered concrete entrance. The second location was an industrial warehouse, where we captured both purely internal footage, as well as footage of transition into the exterior, showcasing the effects of radical change in lighting.

5) Future additions: Our team is actively involved in projects involving flights inside industrial and historical buildings. This will be leveraged to keep the MIDGARD dataset gradually expanding with footage obtained with the proposed system from flights in these environments.

V. Conclusion

In this paper, we have proposed a new method for fast, automatic generation of datasets for training ML methods for visual relative localization of MAVs by other MAVs. The method uses our specialized system incorporating the *UVDAR* system for relative localization, and makes it possible to develop training datasets on the fly, specifically tailored for the needs of an ML application, by deploying desired models of MAVs into arbitrary operational environments. The software used for the processing is based on Robot Operating System (ROS), and is provided on-line. On demand, we can also provide the hardware of the *UVDAR* system.

As an additional contribution, we provide a large dataset called MIDGARD, which was generated using the proposed method. We believe that this dataset will promote the development of ML approaches for practical field deployment of multi-robotic flight systems, and also of MAV systems intended for interaction with other MAVs. The annotated dataset comprises color camera images with MAVs in various environments, with their positions and bounding boxes provided. The dataset, together with a brief video demonstration of the proposed system, is available at `mrs.felk.cvut.cz/midgard`. We are enthusiastic about the possibility of opening cooperation with peers who can use our proposed system, which can potentially yield footage of other unusual MAVs or normally inaccessible flight locations.

References

[1] M. Vrba, D. Heřt, and M. Saska, "Onboard marker-less detection and localization of non-cooperating drones for their safe interception by an autonomous aerial system," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3402–3409, Oct 2019.

[2] M. Vrba and M. Saska, "Onboard marker-less MAV detection and localization using neural networks," *IEEE Robotics and Automation Letters*, 2020, in review.

[3] J. Li, D. H. Ye *et al.*, "Multi-target detection and tracking from a single camera in Unmanned Aerial Vehicles (UAVs)," in *IROS*, 2016.

[4] A. Rozantsev, V. Lepetit, and P. Fua, "Flying objects detection from a single moving camera," in *CVPR*, 2015.

[5] K. R. Sapkota, S. Roelofsen *et al.*, "Vision-based Unmanned Aerial Vehicle detection and tracking for sense and avoid systems," in *IROS*, 2016.

[6] R. Opromolla, G. Fasano, and D. Accardo, "A vision-based approach to UAV detection and tracking in cooperative applications," *Sensors*, vol. 18, no. 10, 2018.

[7] M. Saqib, S. D. Khan, N. Sharma, and M. Blumenstein, "A study on detecting drones using deep convolutional neural networks," in *IEEE AVSS*, 2017.

[8] A. Schumann, L. Sommer *et al.*, "Deep cross-domain flying object classification for robust UAV detection," in *IEEE AVSS*, 2017.

[9] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *CVPR*, 2017.

[10] W. Liu, D. Anguelov *et al.*, "SSD: single shot multibox detector," *CoRR*, vol. abs/1512.02325, 2015.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, June 2017.

[12] Q. Ali, N. Gageik, and S. Montenegro, "A review on distributed control of cooperating mini uavs," *International Journal of Artificial Intelligence & Applications*, vol. 5, pp. 1–13, 07 2014.

[13] D. A. Mercado, R. Castro, and R. Lozano, "Quadrotors flight formation control using a leader-follower approach," in *ECC*, July 2013.

[14] T. Chen, Q. Gao, and M. Guo, "An improved multiple uavs cooperative flight algorithm based on leader follower strategy," in *CCDSC*, 2018.

[15] S. van der Helm, M. Coppola, K. N. McGuire, and G. C. H. E. de Croon, "On-board range-based relative localization for micro air vehicles in indoor leader–follower flight," *Autonomous Robots*, 2019.

[16] M. Saska, T. Báča *et al.*, "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.

[17] M. Saska, "Mav-swarms: Unmanned aerial vehicles stabilized along a given path using onboard relative localization," in *ICUAS*, 2015.

[18] A. Censi, J. Strubel *et al.*, "Low-latency localization by active led markers tracking using a dynamic vision sensor," in *IROS*, 2013.

[19] A. G. *et al.*, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *CVPR*, 2012, pp. 3354–3361.

[20] J. Deng, W. Dong *et al.*, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009, pp. 248–255.

[21] P. Zhu, L. Wen *et al.*, "Vision meets drones: A challenge," *CoRR*, vol. abs/1804.07437, 2018.

[22] T. Krajník, M. Nitsche *et al.*, "A practical multirobot localization system," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 3-4, pp. 539–562, 2014.

[23] V. Walter, M. Saska, and A. Franchi, "Fast mutual relative localization of uavs using ultraviolet led markers," in *ICUAS*, 2018.

[24] 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 *(CASE 2018)*, 2018.

[25] 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, July 2019.

[26] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A flexible technique for accurate omnidirectional camera calibration and structure from motion," in *ICVS*, 2006.

[27] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *IEEE*, vol. 92, no. 3, pp. 401–422, March 2004.

UVDAR System for Visual Relative Localization with application to Leader-Follower Formations of Multirotor UAVs

Viktor Walter[1], Nicolas Staub[1], Antonio Franchi[2] and Martin Saska[1]

Abstract—A novel onboard relative localization method, based on ultraviolet light, used for real-time control of a leader-follower formation of multirotor UAVs is presented in this paper. A new smart sensor, UVDAR, is employed in an innovative way, which does not require communication and is extremely reliable in real-world conditions. This innovative sensing system exploits UV spectrum and provides relative position and yaw measurements independently of environment conditions such as changing illumination and presence of undesirable light sources and their reflections. The proposed approach exploits this retrieved information to steer the follower to a given 3D position and orientation relative to the leader, which may be considered as the main building block of any multi-UAV system operating with small mutual distances among team-members. The proposed solution was verified in demanding outdoor conditions, validating usage of UVDAR in real flight scenario and paving the way for further usage of UVDAR for practical multi-UAV formation deployments.

Index Terms—Aerial Systems: Perception and Autonomy, Multi-Robot Systems, Sensor-based Control

I. INTRODUCTION

THE growing interest in compact cooperative flights of Unmanned Aerial Vehicles (UAVs) [1] motivates an ongoing pursuit for efficient and embeddable onboard source of mutual relative localization.

In our previous work [2], we proposed a novel approach to tackle this issue, relying on vision in the unconventional ultraviolet spectrum. We named this new onboard sensor *UVDAR* for UltraViolet Direction And Ranging, and together with blinking ultraviolet markers used on its associated targets these comprise the *UVDAR system*. Its main advantages w.r.t. other solutions are twofold. First, the use of UV significantly increases robustness to challenges of outdoor environments regardless of the time of day, and second, its use of active markers allows for retrieval of orientation or identity of a
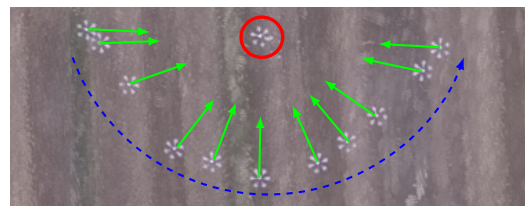
Fig. 1: Top view of a *directed leader-follower* experiment. The leader rotates by $180°$. This is detected by UVDAR sensor carried by the follower, triggering it to create a trajectory as per the proposed algorithm in order to preserve its pose in the leader frame.

target. The availability of such robust sensor is a prerequisite for decentralized outdoor formation flights and swarming and is especially crucial when a sufficiently precise absolute localization source is unavailable, or when it is unfeasible to prepare the necessary infrastructure [3], such as a motion-capture system (MoCap) or a base-station for *Real-time kinematic - Global Navigation Satellite System* (RTK-GNSS).

A typical example of multi-UAV flights is the leader-follower formation, consisting of two members, one following the other. Typically, the goal in such flight is for the follower to keep a constant distance from the leader, or to follow its trajectory [4]. Such following is applicable for various tasks such as cooperative mapping of historical buildings, cooperative carrying of objects or cooperative localization of a moving transmitter [3], [5]. In this paper, we show how the leader-follower approach has to be designed to be able to perform the required behavior using the UVDAR sensor. The presented *directed leader-follower* method, which leverages relative orientation information, can be considered as a guideline for designing complex multi-UAV systems working in real conditions with this sensor.

The literature on classical leader-follower formations is rich, see, *e.g.*, [4], [6], [7] for theoretical works backed by simulation. Works addressing the challenges of real experiments are limited, especially relying on onboard relative localization. The experimentally validated approaches often rely on either absolute localization source, *e.g.*, MoCap in [8], or RTK-GNSS in [9], [10]. As is known, MoCap is not practical for real-world deployment (neither outdoor or indoor) as it requires the installation of an expensive infrastructure. The absolute localization sources can provide full pose of the leader to the follower, which oversimplifies the problem.

CHAPTER 4. RESEARCH MOTIVATED BY THE *UVDAR* SYSTEM

Fig. 2: Comparison between visible and UV camera footage from UVDAR, collected during the experiment. The UV image is significantly easier to process to retrieve UAV information.

Even if only partial information is passed to the follower motion controller, like distance or bearing, this type of system provides continuous stream of such information without errors, which is unrepresentative of real-world deployment.

Some more practical approaches consider infrastructure-less sensing, like ranging based on radio signal [11], which only allows for distance-based following without any orientation information. Another approach [12], for the 2D case, wirelessly communicates the leader intents, which proves feasible since there are less degrees of freedom and less drift than in the presented 3D case. These two approaches rely on radio transmissions, which is subject to the effects of network congestion and interference. This is why we consider vision-based approaches more suitable for the multi-robot groups, especially in uncontrolled outdoor environments.

This direction has been previously explored by the Multi-robot Systems (MRS) group at CTU-Prague, relying on true outdoor relative localization, see [13], [14]. The source of relative localization was an onboard vision-based system using passive circular markers as described in [15]. That came with drawbacks: high sensitivity to lighting conditions and partial occlusion, and substantial size for an acceptable detection range. A similar approach has been proposed [16], that extended the usability of passive markers for low light in short distances by leveraging the infra-red reflection. However, in all other respects it suffers from the same drawbacks as the visible passive marker approach. Furthermore, it was tested only for stable ground vehicles. This motivated the development of the UVDAR system, which is more robust to real-word conditions, due to optically filtering out visual information that is not of interest, reducing the computational load, see Fig. 2 for comparison with visible spectrum. UVDAR also provides relative orientation measurements and target identities, and the whole system is small and lightweight.

Our contribution is threefold. We first show how UVDAR can be used to obtain both the relative position and orientation. We then propose a *directed leader-follower* algorithm that works interactively with the UVDAR sensor and measurement method. Finally, we validate the performance of the presented method in outdoor experiments.

II. POSE RETRIEVAL PRINCIPLE FOR UVDAR

The UVDAR sensor, presented in our previous work [2] [17], retrieves image positions and frequency-based IDs for individual blinking ultraviolet markers from a modified camera. This data is used to obtain the relative pose of the leader.

The blinking markers carried by the leader UAV have a known layout. We found that six markers arranged in a pattern of a regular hexagon pose as a good compromise, that ensures that at least two markers are visible from each direction and the markers not being too close to each other. This means that they provide a source of a distance estimation without their images tending to merge in the operational distances. We instantiate this arrangement on a regular hexarotor platform with the markers attached to the ends of its arms, but the arrangement is easily reproduced for any similar rotorcraft, by *e.g.*, mounting the six markers onto a horizontal ring attached to the vehicle. If a different number of markers was to be used, the calculations used in this section need to be adjusted accordingly. In particular, if the arrangement will result in different number of markers being visible from different directions, each case needs separate equations according to the given geometry. The relative yaw is obtained by giving these markers two distinct IDs, retrievable by the UVDAR, one for the three port side markers and another for the three starboard ones. For other shapes of UAVs, different configurations may be preferable, see [17].

In this section we introduce the calculations necessary to retrieve the relative bearing, the mutual distance and the relative yaw when UVDAR system is used in conjunction with regular distribution hexarotors. The relative bearing is the direction towards the leader in the follower body-frame and the mutual distance is the distance between the geometrical centers of the two UAVs. The relative yaw is the angle between the horizontal components of their connecting line and the *tailing direction*, which is in our case the backwards direction in the leader body-frame.

Note, that for the mutual distance and relative yaw estimation we are assuming a horizontal alignment between the sensor and the target, since the height difference has negligible effect on the presented distance estimates and no effect on the relative yaw estimates

Two basic cases of the UVDAR output occur in practice, see Fig. 3, it either sees simultaneously two markers (case A, Fig. 3a) or three markers(case B, Fig. 3b). This depends on the relative yaw of the leader, because of the Lambertian radiation pattern of the markers, leading to two different calculations to retrieve the values of interest. In both cases the distance is first retrieved based on geometrical considerations and then the relative yaw based on marker IDs.

A. Distance Retrieval – case A

The pixel coordinates $\mathbf{m_i}$ of the origin points (the current expected image position of a blinking marker obtained by UVDAR, see [2]) is first translated into a 3D unit vector $\mathbf{v_i}$ pointing towards the marker,

$$\mathbf{v_i} = \mathrm{c2w}(\mathbf{m_i}), \tag{1}$$

where $\mathrm{c2w}(\cdot)$ is a standard function available in a number of vision libraries, such as the *OCamCalib* toolbox [18], provided that the camera has been properly geometrically calibrated.

(a) Geometry used when two markers are seen (case A)

(b) Geometry used when three markers are seen (case B)
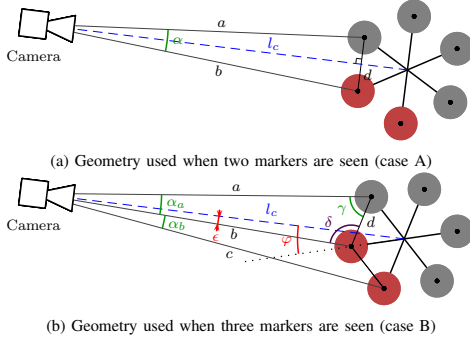
Fig. 3: The notation used in relative UAV pose estimation.

Fig. 4: Left – The layout of the two marker IDs on the considered hexarotor, denoted as blue (6 Hz) and red (15 Hz). Around it the frames illustrate the UVDAR view from their corresponding direction. On the right, the actual view from four different directions from experiments.

As we consider only two markers, the angle their corresponding vectors form is denoted α and obtained via cross-product, $\alpha = \arccos\left(\mathbf{v_1} \cdot \mathbf{v_2}\right)$.

This angle is used to calculate the distance l_c between the target UAV geometrical center and the sensor, while presuming that the line segment between the two markers is perpendicular to the line connecting the camera with its center point. This yields

$$l_c = \left(\frac{d}{2}\right)\cot\left(\frac{\alpha}{2}\right) + \sqrt{0.75\,d}\,, \qquad (2)$$

where d is the length of the hexarotor arm, see Fig. 3a.

The relative bearing vector $\mathbf{v_c}$, is obtained through the conversion in (1) applied on the point in between the two origin points visible. The distance l_c and vector $\mathbf{v_c}$ describe the relative position of the leader w.r.t. the UVDAR sensor.

Note that with only two markers visible, there is an ambiguity on α arising from the simultaneous influence of distance and relative yaw (orthogonality assumption), which is explored in depth in [17]. This ambiguity disappears if three markers are visible.

B. Distance Retrieval – case B

When three markers are visible, see Fig. 3b, the angles α_a and α_b are computed via (1) from two adjacent origin points. The distance l_c and angles φ and ϵ are expressed as

$$l_c = \sqrt{b^2 + d^2 - 2bd\cos\left(\delta + \frac{\pi}{3}\right)} \qquad (3)$$

$$\epsilon = \arcsin\left(\frac{d}{l_c}\sin\left(\delta + \frac{\pi}{3}\right)\right) \qquad (4)$$

$$\varphi = \arcsin\left(\frac{b}{l_c}\sin\left(\delta + \frac{\pi}{3}\right)\right). \qquad (5)$$

Here, b and δ are common terms which correspond respectively to the distance from the sensor to the middle marker and the angle formed by the left marker, middle marker and the sensor. They can be expressed as follows

$$b = \frac{d\sin(\pi - (\delta + \alpha_a))}{\sin(\alpha_a)} \text{ and } \delta = 2\arctan\left(\frac{P}{\sqrt{3}B + 3}\right).$$

The last equation is a compact form of the analytical solution of a set of non-linear equations, where

$$A = \cot(\alpha_a) \quad B = \cot(\alpha_b)$$

$$P = B\left(2\sqrt{\frac{O}{(B^2 + 2\sqrt{3} + 3)}} - 1\right)$$

$$+ \left(6\sqrt{\frac{O}{(\sqrt{3}B + 3)^2}}\right) + (2A + \sqrt{3})$$

$$O = A^2 - AB + \sqrt{3}A + B^2 + \sqrt{3}B + 3.$$

The relative position is then estimated from l_c and the relative bearing $\mathbf{v_2}$ of the middle marker rotated by ϵ in yaw.

C. Relative Yaw Retrieval – case A

The relative yaw in camera frame $\Delta\psi_c$ is retrieved from the IDs of the UVDAR markers. Only a finite number of patterns can be observed, as seen in Fig. 4, which we numbered from I to XII.

When only two origin points are seen, it corresponds to six possible relative orientations. If the two IDs differ, the leader is seen either from the front (XII and $\Delta\psi_c = \pi$) or from the back (VI and $\Delta\psi_c = 0$), *i.e.*, the tailing direction. If the IDs are identical, the orientation is ambiguous (II–IV and VII–X). We resolve this with an heuristic, by averaging the two possible interpretations of such observation. Namely, $\Delta\psi_c = \pm\pi/3$ and $\pm 2\pi/3$, so the average is $\Delta\psi_c = \pi/2$ on starboard side or $-\pi/2$ on port side. Note, that resolving the ambiguity based on previous observations is precluded by the ability of the target to independently change its rotation rate at any moment.

D. Relative Yaw Retrieval – case B

When three origin points are seen, we consider the other six possible relative orientations, see Fig. 4. They correspond with relative orientation s.t. the follower is roughly facing one specific arm ($\Delta\psi_e = \pm(\pi/6 + k(\pi/3))\colon k \in \{0, 1, 2\}$).

CHAPTER 4.   RESEARCH MOTIVATED BY THE *UVDAR* SYSTEM
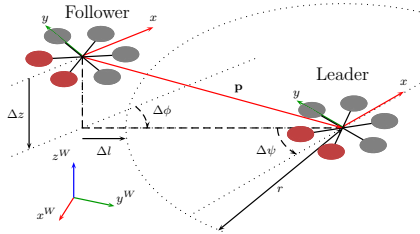
Fig. 5: Variables defining our requirements on the leader-follower system. Red propellers denote the back of the UAV. We want the leader to be on the circular perimeter with the radius r around the leader ($\Delta l \to 0$) at the height of the leader ($\Delta z \to 0$), while facing it ($\Delta \phi \to 0$) from its back ($\Delta \psi \to 0$).

(a) Following

(b) Orbiting

(c) Flanking

(d) Retreating

Fig. 6: The four strategies used in our directed leader-follower experiment. The red propellers denote the back of the UAV.

Imperfect sensor alignment with the corresponding arm is accounted for by using the angle φ obtained from (5) as $\Delta \psi_c = \Delta \psi_e - \varphi$. Note that typically, for the considered operational distances greater than $5\,\text{m}$ the origin points are very close in the UVDAR image, which means that α_a and α_b are almost identical, and ϵ and φ had negligible effects.

In both cases, before the information retrieved from the UVDAR, *i.e.*, estimates of the relative position and yaw are used in our *directed leader-follower* they are transformed into the IMU frame of the follower UAV, which compensates not only for the offset of the sensor but also for the tilt (*i.e.*, roll and pitch) of the unit. Thus corrected relative yaw is denoted as $\Delta \psi$, and is used in the following section as *tailing error*.

III. DIRECTED LEADER-FOLLOWER

The goal of *directed leader-follower* formation is to simultaneously regulate the mutual distance towards a pre-set *tailing distance* and to let the follower always face a given leader side (*tailing direction*), *e.g.*, its back. Our proposed algorithm solves such task and takes also into account constraints of vision-based sensing: 1) forward facing directional sensor and 2) sensitivity to rapid image motion. To address both, the follower behavior is such that it always attempts to face the leader. This guarantees that the leader is in the field of view (FoV) of the sensor and additionally that the leader image position will not change greatly over short periods of time. The requirements of the system are therefore to minimize the control errors illustrated in Fig. 5 as Δl - the distance error, Δz - the height error, $\Delta \phi$ - the heading error and $\Delta \psi$ - the tailing error. These are equal to zero if the follower is in what we call the *target pose* w.r.t. the leader. The heading error is the horizontal angle between the bearing of the leader and the frontal direction of the follower.

A. Trajectory Generation Strategies

The goal is to steer the follower to the target pose, located on the back of the leader, at a distance r, *tailing distance*, by which we also define a safety perimeter around the leader.

If the follower is steered only with the currently observed leader pose, changes in the observation lead to rapid changes
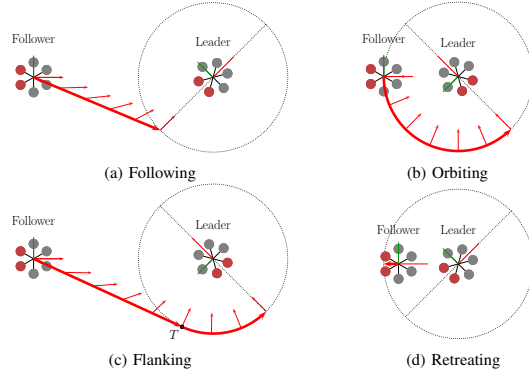
in the follower target pose. This is detrimental as we consider under-actuated platforms which have high coupling between their translational and rotational dynamics. Hence higher translational acceleration means higher tilting, likely to perturb the visual localization. To avoid this, we design the algorithm such that it repeatedly constructs a short-term trajectory, at fixed rate, whose time horizon is at most $4\,\text{s}$. The trajectory consists of isochronous points defined by their position and yaw. Trajectories are naturally constructed in the follower body-frame, if the follower is localized in the world-frame it is possible to convert them to world-frame to accommodate for low-level trajectory trackers.

The trajectory is constructed according to one of four distinct strategies; 1) following, 2) orbiting, 3) flanking, and 4) retreating, as depicted in Fig. 6.

For each strategy we consider the height error and the lateral position errors separately, as the height error does not play into the trajectory selection, and we attempt to bring it to zero as fast as possible in all four cases. We do this, by setting the **z** component of the whole generated trajectory directly to the **z** component of the estimated relative position of the leader, which forces our trajectory tracker to reach this height as fast as it can, bringing Δz close to zero.

The appropriate lateral strategy is selected based on the current situation which is described by the tailing error $\Delta \psi$, the tangential angle $\beta = \arccos \left(r / l_c \right)$ and the distance error Δl, see Fig 7. The decision map is as follows:

$$\Delta l < -h \qquad\qquad\qquad\qquad \to \text{Retreating} \qquad (6)$$

$$\Delta l \in [0, -h] \qquad\qquad\qquad \to \text{Orbiting} \qquad (7)$$

$$(\Delta l > 0) \ \wedge \ (|\Delta \psi| < |\beta|) \qquad \to \text{Following} \qquad (8)$$

$$(\Delta l > 0) \ \wedge \ (|\Delta \psi| \geq |\beta|) \qquad \to \text{Flanking} \qquad (9)$$

where h is a tolerance factor, introduced to prevent rapid switching in boundary cases by creating some hysteresis.

In the *following* strategy, Fig. 6a, the follower flies directly to the target pose on the perimeter at its maximum admissible horizontal speed.
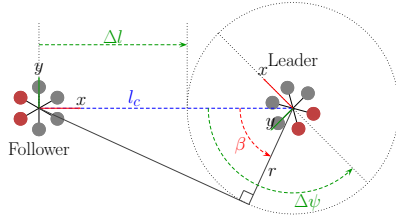
Fig. 7: Parameters of the observed situation for strategy selection.

In the *orbiting* strategy, Fig. 6b, as the follower is already at the tailing distance, it slides along the perimeter to reach the desired tailing direction.

In the *flanking* strategy, Fig. 6c, the follower flies straight to the tangent point T of the perimeter closer to the target pose, which brings the follower in the *orbiting* configuration.

In the *retreating* strategy, Fig. 6d, the follower is inside the safety perimeter of the leader, hence it is navigated outside of it radially, at its maximum admissible horizontal speed.

These strategies are devised to generate a fast path toward the target pose that is continuous up to the first derivative. Generation of each trajectory is based on the measured relative poses of the leader and the follower, from which estimates of the safety perimeter and the target pose lying on it are calculated. Depending on the current strategy, linear, circular or a combined trajectory is generated. The employed sampling of the trajectory ensures that the euclidean distance between two consecutive positions corresponds to the distance traversed at the maximum admissible speed during a single time-step, enforcing constant tangential speed. In order to accommodate for a leader motion, strategy selection and trajectory generation are triggered at a fixed rate.

Additionally, for each strategy we enforce that the follower yaw is such that its camera always faces the estimated leader position, considering the error $\Delta\phi$, by setting the reference yaw in each step of the trajectory to face the currently estimated leader position. This ensures continuous observation without rapid movements in the image as well as preventing loss of the leader from view in case of limited FoV, in our case $180°$ in the horizontal axis.

The generated trajectories are not accounting for real-world dynamics of the UAV and should be filtered before being sent to the low-level trajectory tracker. In our experimental setup, we leveraged the model predictive control present in our system [19], making the final trajectory smooth. This alters the original trajectory, but the optimization procedure used in [19] minimizes these differences, so that the resulting trajectory differs from the original only in four specific cases. Firstly, at the start of the trajectory after the leader was first discovered, the follower first accelerates to reach the desired tangential speed. This does not happen if the leader was already being tracked, since in such case the initial state already includes the tangential speed. Secondly, if the target pose is reached within a single trajectory generation period, the follower will decelerate, since abrupt stopping is unfeasible. The third situation

occurs during the transition between the linear and circular phase of the flanking strategy, when the trajectory is adjusted to achieve continuous acceleration. In this case the resulting trajectory resembles turns in automobile roads, eliminating step change in acceleration. This result is possible, because the flanking strategy contains both phases. An additional benefit is that if the next strategy is orbiting, after reaching the perimeter, the initial state will already include appropriate tangential speed so that the original trajectory will be followed with minimal change. Lastly, when retreating the trajectory is set such that the follower retreats according to its maximum speed, without regards to other conditions, which the model predictive control interprets by applying the maximum admissible acceleration. As the trajectory is re-generated asynchronously, following one of the four policies, the current state of the model is fused with the new trajectory to ensure a smooth transition. One useful addition for initialization of the leader-follower task or if the leader is lost, is setting the follower to slowly spin in place if it has not detected the leader yet, or has not seen it for pre-defined time.

B. Constraints on the leader motion

In order for the follower not to lose the leader and to prevent collisions, the motion of the leader must conform to a set of restrictions.

The blinking signal retrieval in UVDAR limits the maximum component of the marker velocity perpendicular to the associated camera optical ray, in order to ensure consistent tracking. With our typical frame-rate of *approx.* $72\,\mathrm{Hz}$, $23\,\mathrm{frame}$ signal sample and maximum allowed marker shift between frames of *approx.* $1\,\mathrm{pixel}$, this limit is $0.3 * l\,\mathrm{m\,s^{-1}}$, scaling with the real distance l between the UAVs. The limit also defines the maximum yaw rotation rate of the leader, corresponding to *approx.* $0.3 * l/d\,\mathrm{rad\,s^{-1}}$. Additionally, linearity assumption in the UVDAR [2] limits the maximum acceleration of the leader in this direction to $0.3 * l\,\mathrm{m\,s^{-2}}$.

While tracking is unaffected by the component of the velocity along the camera optical rays, the distance measurement is less precise than the relative bearing. In particular, earlier experiments [17], showed that in rare conditions the distance measurement error could get close to $20\,\%$.

The distance estimate is important for the follower to successfully retreat from the leader in case of breached perimeter. This breach must be detected in time despite the distance estimate possible error. In the adverse case where both UAVs are flying directly towards each other, the follower at its following speed of v_{Fmax} and the leader at v_L, the follower registers a perimeter breach with delay. Additionally, this delay is extended by filtering the distance estimate with a moving average filter of time window t_a, when the detection is delayed by $t_d = t_a/2$. Note that the detriments of the moving average filter in this case are balanced by enhanced performance of the bearing estimate. The perimeter breach is detected at the distance

$$l_{brake} = (r - h) * 0.8 - r_{coll} - (v_L + v_{Fmax})t_d,$$

CHAPTER 4. RESEARCH MOTIVATED BY THE *UVDAR* SYSTEM
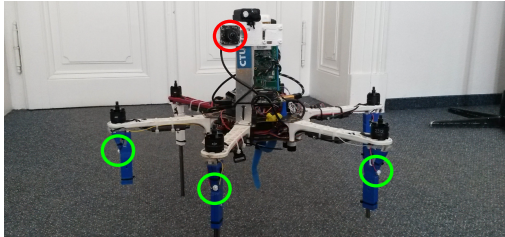
Fig. 8: The UAV platform used in our experiments, here equipped both with a ultraviolet camera (red) and active markers (green) comprising the hardware components of UVDAR system.

Fig. 9: Top view of the leader and follower trajectories in the preliminary experiment without marker identity. For a sense of time, selected simultaneous positions are marked with the same color.

where r_{coll} is the collision distance. If a_{Fmax} is the maximum feasible acceleration of the follower in the case of retreating, then collision in the worst case can be avoided if

$$v_L < \sqrt{2l_{brake}a_{Fmax}} - v_{Fmax},$$

when $l < r$. In our experimental setting, this translates to limiting the approaching speed to $v_L < 0.61\,\mathrm{m\,s^{-1}}$ for distances smaller than $r = 5\,\mathrm{m}$. A violation of this restriction is shown in the red zone of Fig. 10, resulting in the follower not retreating fast enough.

Evidently for greater safety, the leader should avoid approaching the follower. The rough direction in which the follower lies is implicitly known to the leader, since the follower is set to face a specific side of the leader. If the leader needs to fly in this direction, a simple way to prevent approaching the follower is to first rotate, ideally by $90°$, and thus to steer the follower out of the way into a relative pose from which it can easily follow in a sideways manner.

Lastly, since the maximum distance for reliable detection by the UVDAR is $15\,\mathrm{m}$, the leader, when it is further than $12.5\,\mathrm{m}$ from the follower, must not retreat from it faster than $v_{Fmax}\ \mathrm{m\,s^{-1}}$. This will ensure that the error of distance measurement will not lead to the follower losing the leader from sight. In most cases, the following algorithm already accounts for this, if the following distance is set to less than $12.5\,\mathrm{m}$ and enough time is provided for the follower to reach the target pose at the start of the mission.

IV. Outdoor Experimental Validation

A. Experimental Platform

In order to validate the performances of the proposed formation algorithm, we conducted a campaign of real-world outdoor flights with two *DJI f550*-based hexarotors, see Fig 8. They are each fitted with a *Intel NUC7* computer, a *PixHawk* flight controller and a *Tersus* GPS receiver, used with a RTK-GNSS system to obtain ground truth.

The two units were each equipped with a part of the UVDAR system. The leader was equipped with ultraviolet markers attached on the ends of its arms. The markers can be set with a blinking ID or not. In our setup two IDs are used as depicted in Fig. 4. Apart from providing IDs, blinking markers ensure robustness against reflections of the sun.
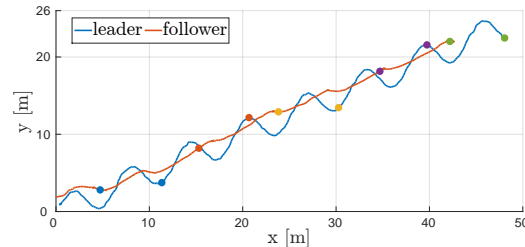
The follower was equipped with a front facing UVDAR sensor, with a fisheye lens, allowing for $180°$ of horizontal FoV. Resolution and typical frame-rate are $752 \times 480\mathrm{pix}$ and $72\,\mathrm{Hz}$, respectively. With the current UVDAR settings the detection range is around $15\,\mathrm{m}$, see [17]. The relative positions and yaws of the leader are cyclically estimated at the rate of $10\,\mathrm{Hz}$.

In order to increase the precision and to suppress the effect of any spurious errors of detection on the flight, we use a moving average filter of window 10, on the relative distance and relative yaw estimate. The relative bearing does not need filtering, as it is derived from the image position of the target, which we consider to be sufficiently reliable and precise.

During the experiments we noted that the UVDAR is highly sensitive to the lens focus. Indeed repeated manipulation altering the focus made the detection range drastically decrease to around $6\,\mathrm{m}$, insufficient for practical purposes. Fortunately, focus can be monitored and adjusted easily.

For visualization, comparison and future analysis, the follower also carried a front-facing color camera. The views from the two onboard cameras are recorded a low frame-rate, so as not to impede the UVDAR.

B. Preliminary Flight – Without Marker Identity

In a first set of experiments, we validated the UVDAR performances as a distance and relative bearing sensor, before going further. To do so, a simple leader-follower formation was tested. The markers of the leader were not blinking and the follower set to only approach the leader up to a desired tailing distance. This was implemented as a simple proportional position controller. Such behavior has been demonstrated with various other sensors. The distance and relative bearing are obtained as described in Sec II.

The leader tracked a waving trajectory retreating from the follower, see Fig. 9. The follower successfully managed to tail the leader during the whole trajectory, of length $214\,\mathrm{m}$, demonstrating that UVDAR provides sufficient distance and relative bearing measuring capabilities for real-world flight.

C. Real-World Flight of the Directed Leader-follower

Since the UVDAR is also able to provide a useful relative orientation estimate, we have conducted a second real-world

(a) The trajectory of the leader

(b) Follower control errors
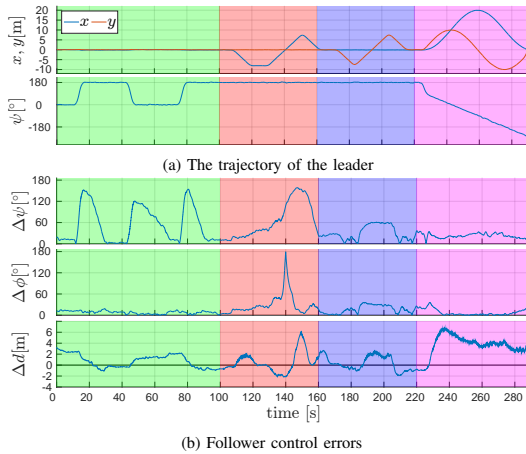
Fig. 10: (a) The leader trajectory defined by position and yaw ψ. The height was constant, set to $8\,\mathrm{m}$. (b) From top to bottom - tailing error, heading error and distance error. Colored zones correspond to specific leader motions - rapid rotation (green), retreat-approach (red), left-right (blue), and circle following (purple).

flight where this information is used.

The markers on the leader were set to blink, at 6 and $15\,\mathrm{Hz}$ following the pattern depicted in Fig. 4. The distance, relative bearing and orientation are obtained as described in Sec II, and used in the *directed leader-follower* from Sec. III. The tailing distance was set to $r = 5\,\mathrm{m}$, with maximum hysteresis $h = 2\,\mathrm{m}$. The temporary trajectories are generated at $2\,\mathrm{Hz}$ and their tangential speed is a kept at $2\,\mathrm{m\,s}^{-1}$.

Before the leader started its trajectory, it waited for the follower to reach its target pose. The leader trajectory was devised to highlight the system behavior in four representative cases. First, the leader makes three rapid rotations in yaw by $180°$ with $30\,\mathrm{s}$ of static hovering in between. Second, the leader moves linearly at $0.8\,\mathrm{m\,s}^{-1}$ with static yaw, going forward $8\,\mathrm{m}$, backward $16\,\mathrm{m}$ and then forward $8\,\mathrm{m}$ again.

Third, the same retreat-approach motion was performed from left to right. The fourth case was a car-like following of a circular trajectory with a radius of $10\,\mathrm{m}$. The height of the leader was fixed to $8\,\mathrm{m}$ for the whole experiment. The leader motion and follower control errors are plotted for the full trajectory, with the four cases, in Fig. 10a.

Video of the experiment can be seen online[1] and an external view of the experiment is shown in Fig. 11.

1) Rapid rotation: This highlights the importance and usage of relative orientation. Every time the follower detected a change in the leader orientation, it flew around the leader, see Fig. 1, to successfully reach the target pose again, demonstrating that relative yaw retrieval with UVDAR is reliable enough for real-world applications.

[1]http://mrs.felk.cvut.cz/directed-following-with-uvdd

Fig. 11: View of *directed leader-follower* experiment, the leader (red) is retreating from the follower (green).

2) Retreat-approach: This can be seen as a classical leader-follower formation. The follower uses the relative position estimate to maintain a set distance from the leader.

The observed performance is good overall. However, note that due to the granularity of the distance estimate from vision, combined with observation averaging, the reaction of the follower can be delayed, see Fig. 10b, around the $140\,\mathrm{s}$ mark. This engaged the follower collision avoidance mechanism, see [19], forcing it to fly over the follower and then to resume directed following by turning around and orbiting, which is the origin of the observed peak in $\Delta\psi$. This demonstrates that the good following performance can be jeopardized if the leader flies towards the follower faster than the admissible limit of $0.61\,\mathrm{m\,s}^{-1}$ estimated in III-B, since in this case we set the speed of approach to $0.8\,\mathrm{m\,s}^{-1}$. A larger perimeter can be set to mitigate this, trading off visual distance estimation precision, so a compromise needs to be found for each application.

3) Left-right: When the leader moves side to side, both the relative distance and orientation estimates performances are evaluated. As depicted in Fig. 10b the performances are good as the max. relative yaw was around $60°$ and the distance error around $2\,\mathrm{m}$, demonstrating that the follower was able to deal with a continuous disturbance in both quantities simultaneously.

4) Circular following: The last part of the trajectory demonstrated the ability of our system to follow a leader along an extended trajectory, by tailing a leader flying along a circular trajectory in a car-like manner. As the plot implies, while the follower lagged behind the tailing distance by $4\,\mathrm{m}$ on average, it did not lose track of the leader for the whole trajectory, both in terms of relative yaw and heading.

V. POTENTIAL FUTURE EXTENSIONS

The system performances in the experiments validated our approach and more importantly pave the way to a wider use of UVDAR for multi-UAV relative localization.

Other formations can be explored and tested, such as train-like formation where multiple units are following another one in front of them. Thanks to the marker IDs provided by the UVDAR, that can map to leader identities, keeping the leader-follower order should prove easy. Such formation needs to guarantee that leader motion can not force followers further down the line to reach speed or acceleration limits.

Additional studies on leveraging the marker layout are necessary. The current layout, with adjacent triplets of same-ID markers has the drawback that for a pair of same-ID (Fig. 4–II,IV,VIII,X), the relative yaw is ambiguous, *e.g.*, II and IV appear identical. In Sec. II-D we used a heuristic, averaging the two possible interpretations. Another available option is presuming one of the two interpretations based on which leads to the more favorable dynamics. With the current layout, leader starboard and port directions (III and IX) can not be chosen as alternative tailing directions since they are surrounded by ambiguous observations.

However, with the current layout it is possible to steer a follower to any of the other alternative tailing directions that can be uniquely located (Fig. 4-V,VI,VII,XII,XI,I). This allows for multiple directed followers for a single leader, separated by different tailing directions. A simpler way to allow for more followers is to assign them to different relative heights, although the aerodynamic coupling between the followers must be taken into account in that case.

For steering towards one of the unique positions the current layout was sufficient, but for truly arbitrary static formations a third identity must be introduced, using up more of the limited number of available IDs.

Filtering distance estimation with simple averaging proved to be a weak point, imposing strong motion restriction on the leader, and more advanced filtering techniques, such as a Kalman filter, should be considered in the future.

VI. Conclusion

In this paper, we demonstrated the applicability of our novel vision-based relative localization system UVDAR for cooperative UAV flights on a specific implementation of the leader-follower formation. This *directed leader-follower* formation control exploited the relative leader pose obtained by the UVDAR sensor, comprising position and yaw, to steer the follower to a *target pose* pose w.r.t. the body of a moving leader, while also preserving the conditions for continued observation by this vision system. The cooperative combination of UVDAR with a specialized control algorithm was shown to maintain the desired following behavior, without direct communication between the two UAVs.

The encouraging performance of the system for various motions is shown through outdoor experiments. In particular, the use of UVDAR for a real application is demonstrated for the first time, in demanding outdoor situations. More complex formations have to be addressed in future work. Overall, the UVDAR performance in outdoor conditions should lead to its wider adoption.

References

[1] S. Chung, A. A. Paranjape, P. Dames, S. Shen, and V. Kumar, "A survey on aerial swarm robotics," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 837–855, Aug 2018.

[2] 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 *(CASE 2018)*, 2018.

[3] M. Saska, V. Kratký, 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 *ETFA*. IEEE, 2017.

[4] Y. Zou and Z. Meng, "Leader-follower formation control of multiple vertical takeoff and landing uavs: Distributed estimator design and accurate trajectory tracking," in *(ICCA 2017)*, July 2017, pp. 764–769.

[5] M. Saska, "Large sensors with adaptive shape realised by selfstabilised compact groups of micro aerial vehicles," in *International Symposium on Robotic Research*, 2017.

[6] A. S. Brandão, I. H. B. Pizetta, M. Sarcinelli-Filho, and R. Carelli, "High-level nonlinear underactuated controller for a leader-follower formation involving a miniature helicopter and a ground robot," in *(SBR-LARS 2012)*, Oct 2012, pp. 168–173.

[7] J. Choi and Y. Kim, "Fuel efficient three dimensional controller for leader-follower uav formation flight," in *(ICCAS 2007)*, Oct 2007, pp. 806–811.

[8] D. A. Mercado, R. Castro, and R. Lozano, "Quadrotors flight formation control using a leader-follower approach," in *(ECC 2013)*, July 2013, pp. 3858–3863.

[9] S. Montenegro, Q. Ali, and N. Gageik, "A review on distributed control of cooperating mini uavs," 2015.

[10] J. Vilca, L. Adouane, and Y. Mezouar, "Adaptive leader-follower formation in cluttered environment using dynamic target reconfiguration," in *Distributed Autonomous Robotic Systems*, 2016, pp. 237–254.

[11] T. Chen, Q. Gao, and M. Guo, "An improved multiple uavs cooperative flight algorithm based on leader follower strategy," in *(CCDSC)*, June 2018, pp. 165–169.

[12] T. Bhavana, M. Nithya, and M. Rajesh, "Leader-follower co-ordination of multiple robots with obstacle avoidance," in *(SmartTechCon 2017)*, Aug 2017, pp. 613–617.

[13] M. Saska, T. Báča, J. Thomas, J. Chudoba, L. Preucil, T. Krajník, 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.

[14] V. Spurný, T. Báča, and M. Saska, "Complex manoeuvres of heterogeneous mav-ugv formations using a model predictive control," in *(MMAR 2016)*. IEEE, 2016, pp. 998–1003.

[15] T. Krajník, M. Nitsche, J. Faigl, P. Vaněk, M. Saska, L. Přeučil, T. Duckett, and M. Mejail, "A practical multirobot localization system," *Journal of Intelligent & Robotic Systems*, vol. 76, no. 3-4, pp. 539–562, 2014.

[16] H. Park, I. Choi, S. Park, and J. Choi, "Leader-follower formation control using infrared camera with reflective tag," in *(URAI 2013)*, Oct 2013, pp. 321–324.

[17] V. Walter, M. Saska, and A. Franchi, "Fast mutual relative localization of uavs using ultraviolet led markers," in *(ICUAS 2018)*, 2018.

[18] D. Scaramuzza, A. Martinelli, and R. Siegwart, "A flexible technique for accurate omnidirectional camera calibration and structure from motion," in *(ICVS'06)*, 2006.

[19] T. Baca, D. Hert, G. Loianno, M. Saska, and V. Kumar, "Model predictive trajectory tracking and collision avoidance for reliable outdoor deployment of unmanned aerial vehicles," in *(IROS 2018)*. IEEE, 2018.

1

Distributed formation-enforcing control for UAVs robust to observation noise in relative pose measurements

Viktor Walter*, Matouš Vrba*, Daniel Bonilla Licea*, Matej Hilmer*, and Martin Saska*

Abstract— A technique that allows a formation-enforcing control (FEC) derived from graph rigidity theory to interface with a realistic relative localization system onboard lightweight Unmanned Aerial Vehicles (UAVs) is proposed in this paper. The proposed methodology enables reliable real-world deployment of UAVs in tight formation using real relative localization systems burdened by non-negligible sensory noise, which is typically not fully taken into account in FEC algorithms. The proposed solution is based on decomposition of the gradient descent-based FEC command into interpretable elements, and then modifying these individually based on the estimated distribution of sensory noise, such that the resulting action limits the probability of *overshooting* the desired formation. The behavior of the system has been analyzed and the practicality of the proposed solution has been compared to pure gradient-descent in real-world experiments where it presented significantly better performance in terms of oscillations, deviation from the desired state and convergence time.

I. INTRODUCTION

Tight cooperation of multiple robotic Unmanned Aerial Vehicles (UAVs) sharing operational space requires these units to be able to obtain localization of UAVs in their proximity. This information is necessary both for low-level tasks such as collision avoidance, as well as for enforcing a specific layout [2] or for stabilization of the desired shape of a formation. Recent technological developments have rapidly progressed towards the possibility of accomplishing cooperative robotic tasks in unstructured environments, such as arbitrary outdoor spaces or underground tunnels. While in limited cases the UAVs can obtain mutual localization from sources of precise global localization, e.g. Real-Time Kinematic - Global Navigation Satellite System (RTK-GNSS) or Motion capture (mo-cap) systems, this is extremely limiting for a wide range of applications where such a source is unavailable or not reliable enough. Typically, these systems require operators to directly access the operational space ahead of robotic deployment in order to prepare the equipment. Additionally, these are sensitive to environmental conditions and they require extensive radio communication.

The alternative to global localization infrastructure is to use onboard sensors for mutual relative localization [2], [3]. The two most common modalities of these sensors are distance-based relative localization, which is typically based on the strength or timing of a selected omnidirectionally broadcasted

Fig. 1: An example of our fully autonomous Unmanned Aerial Vehicles (UAVs) [1] equipped with the mutual relative localization system UltraViolet Direction And Ranging (*UVDAR*). The units depicted are based on the *MRS F450* platforms. These devices can cooperate using a mutual relative localization and control scheme, such as the one proposed here. The output of the relative localization system is subject to observation noise expressed in terms of covariance of a multivariate Gaussian distribution that is taken into account in the presented method. An illustration of 3D ellipsoids representing the distribution of noise in relative position measurement is shown on the right.

signal [4], [5], and vision-based relative localization using camera systems and image processing. The latter typically provides accurate information on relative bearing with the option of obtaining information on distance and relative orientation with less precision [6]–[14].

Our team has developed an onboard vision-based system for mutual relative localization called UltraViolet Direction And Ranging (*UVDAR*) (visible in Fig. 1). This system can operate robustly both indoors and outdoors with challenging conditions (e.g. arbitrary lighting) thanks to its use of emitters and cameras operating together in a shared narrow band of the ultraviolet (UV) spectrum [14], [15]. The system is capable of retrieving the relative position and relative orientation of marked UAVs [16]. Each of these values is provided with the covariance of its observation noise [17]. The system estimates the distribution of the observation noise using a multi-hypothesis reprojection method for each individual measurement, and the retrieved estimates are processed with a Kalman filter before being used in control. Additionally, the system can provide robotic units with unique retrievable identities, and even possesses limited data communication capabilities without radio transmission [18], [19]. If the odometry of each agent is shared across an explicit communication channel, the precision of the overall relative localization can be improved by fusing this information. However in this paper, we focus on the more general and demanding case where no such communication is taking place. Using this type of explicit communication between numerous UAVs is often challenging in practice and makes the cooperating UAVs susceptible to effects such as network congestion and external interference.

A concurrent research direction addresses new challenges connected with the rise of mutual relative localization systems. Particularly relevant to the current topic is the development of mathematical methods that allow cooperating robots to retrieve, as well as to directly enforce, specific formations in space [20]–[24]. Formation control can also be dynamic, steering agents in constant motion towards a specific shape [25].

2

Formation control can be de-centralized [26], which is advantageous for stand-alone robotic systems where reliance on external infrastructure is problematic.

The paradigm we base our current research on is built on the graph rigidity theory [27]. In this work, we will use the term *grap rigidity* to denote the rigidity of the framework associated with an observation graph formed by inter-agent relative pose measurements.

It has been shown that desired formation can, in theory, be achieved merely based on the mutual relative localization of other units if the localization information is only either the relative distance [28] or the relative bearing [21], [29]. Using nomenclature from [29], all mutual relative measurements within a team form a graph. A relative localization-based formation is said to be rigid if there is no more than one formation that can generate a given set of observations.

In the proposed technique, we will consider the relative measurements of multiple neighboring units that can be provided by the *UVDAR* system. This includes the 3D relative position and associated 1D relative orientation comprising the *relative pose measurement*, which combines and supersedes the relative bearing and relative distance measurements. The relative pose-based formation is always rigid, as long as the observation graph is connected. This is because each single relative pose observation in $\mathbb{R}^3 \times \mathbf{S}^1$ already represents a rigid connection. Thus, any two nodes in an observation graph that are connected through a series of such observations are connected rigidly as well. However, the information provided by real sensors is burdened by observation noise, typically with known statistical properties. This is rarely addressed in the theoretical research of formation control, although considering this apriori knowledge could significantly increase the reliability of multi-robot systems, thereby addressing the main bottleneck of their applicability.

Applying real relative pose measurements into a formation-enforcing control (FEC) derived from the graph rigidity theory while assuming the measurements to be precise leads to undesirable changes in velocity with each new measurement due to noise. Such rapid accelerations and decelerations perturb the vision-based localization system - particularly on lightweight UAV platforms without gimbal or full 6 DoF actuation - by inducing tilting leading to large shifts in the output camera image, blurring, loss of tracking, and targets moving outside of the camera's field of view. These issues feed back into the control in a manner that is difficult to predict. Besides these detrimental effects on measurements, the sensory noise leads to persistent chaotic oscillations of agents around the desired formation. This is especially significant when the relative measurements are obtained in discrete times, since an observation error will affect the behavior of a formation for the entire period before the next measurement is taken. The aim of the presented work was to develop a sensor-based FEC robust to such observation noise.

Literature dealing with imperfect sensing in the context of formation control is sparse. In [30], the authors propose a theoretical distributed formation control that is shown to be robust to bounded measurement errors in the sense that it converges towards a formation close to the desired one. However, the
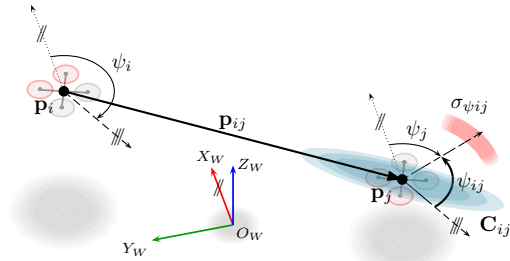
Fig. 2: Measures involved in the construction of the formation for the presented case. Lines marked with ∥ or ⫽ are parallel to those with the same marker.

interdependence of noise-induced parasitic motion with the sensing itself is not addressed, nor does the work attempt to actively counteract the influence of the measurement errors by leveraging the knowledge of their distribution. Various works [31], [32] show stability of FEC under specific singular disturbances. However these assume the control to be a continuous system, in addition to not considering disturbances in the form of incessant noise.

In prior works [33], [34], if the relative observation noise is taken into account, it is usually addressed through linear weighting w.r.t. the distance from the desired values. Not only does this fail to prevent oscillations in states close to the desired formation, but it also slows down the convergence to formation, since such modification alters the motion velocity. This is the case even when the state is far from the desired formation, where the noise represents only a minor component of the FEC action and the agents could safely move faster.

In this paper, we propose a novel distributed formation-enforcing action derived from the graph rigidity theory. This action can mitigate the effects of noise in vision based relative localization on the agent motion, without needlessly sacrificing the speed of convergence to the desired formation. The main contributions of this work can be summarized as follows:

- A novel technique for constructing robust distributed FEC for autonomous UAV teams using onboard relative localization sensors burdened with noise.
- Theoretical analysis of this technique, allowing the user to predict its performance under a given configuration.
- Experimental verification results, showing the performance of the proposed technique in real-world scenarios.
- Publicly available source code of an implementation of the proposed technique.

II. FORMATION-ENFORCING CONTROL

In this section, we will derive a formation-enforcing action for N UAVs with relative pose measurements of their neighbors. We will be using steps similar to [22], where the formation controller was derived for platforms that can measure only the relative bearings of their neighbors. Our case differs, since each individual observation in our case represents a rigid formation in and of itself, the whole formation is rigid as long as the observation graph is connected.

We first denote the world coordinate axes as $(O_W X_W Y_W Z_W)$, where O_W is the origin of the coordinate

3

frame. The horizontal plane $(O_W X_W Y_W)$ is parallel to the earth surface and the $O_W Z_W$ axis points vertically in the opposite direction to gravity. The pose of ith agent in the world coordinate frame is

$$\mathbf{q}_i \triangleq \begin{bmatrix} \mathbf{p}_i \\ \psi_i \end{bmatrix}, \quad i = 1, 2, \cdots, N, \tag{1}$$

where $\mathbf{p}_i \in \mathbb{R}^3$ is the position of the ith agent and $\psi_i \in \mathbf{S}^1$ is its orientation measured as the angle between the $O_W X_W$ axis and the projection of the front-back axis of an agent onto the horizontal plane. We assume the ith agent follows a first-order integrator dynamic model

$$^i\dot{\mathbf{p}}_i = \mathbf{u}_i, \tag{2}$$
$$^i\dot{\psi}_i = \omega_i, \tag{3}$$

where $\mathbf{u}_i$ and ω_i are the control signals for the velocity and the heading speed respectively and $^i\cdot_i$ denotes that the rate is expressed in the body frame of agent i. We also define the formation vector

$$\mathbf{q} \triangleq \begin{bmatrix} \mathbf{q}_1^T, \mathbf{q}_2^T, ..., \mathbf{q}_N^T, \end{bmatrix}^T. \tag{4}$$

The noiseless relative pose of the agent j w.r.t. the agent i in the local frame coordinate of the agent i is

$$\mathbf{q}_{ij} = \begin{bmatrix} \mathbf{p}_{ij} \\ \psi_{ij} \end{bmatrix} = \begin{bmatrix} \mathbf{R}(\psi_i)^T(\mathbf{p}_j - \mathbf{p}_i) \\ \psi_j - \psi_i \end{bmatrix}, \tag{5}$$

$$\mathbf{R}(\psi_i) = \begin{bmatrix} \cos(\psi_i) & -\sin(\psi_i) & 0 \\ \sin(\psi_i) & \cos(\psi_i) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{6}$$

Whenever angles are subtracted, the result is unwrapped so that it lies within $[-\pi, \pi]$. This applies for all such occurrences in this paper, but to simplify the text, we will not repeat this again.

We define the set containing all the relative poses observed by the agent i as:

$$\mathcal{Q}_i \triangleq \{\mathbf{q}_{ij} : j = 1, \cdots, N, \quad j \neq i, \quad c_{ij} = 1\}, \tag{7}$$

where $c_{ij} = 1$ if agent j is observed by agent i, and $c_{ij} = 0$ otherwise. Observations ij such that $c_{ij} = 1$ form the edges of an *observation graph* $\mathcal{G}$. We also define the set of all observed relative poses as

$$\mathcal{Q} \triangleq \cup_{i=1}^N \mathcal{Q}_i. \tag{8}$$

To obtain the formation-enforcing action, we need to first define the *relative pose function* $\varkappa_{\mathcal{G}}$ that transforms the world poses $\mathbf{q}$ into a set of observed relative poses of neighbors in the body frame of each observing agent as

$$\varkappa_{\mathcal{G}}(\mathbf{q}) = \begin{bmatrix} [\mathcal{Q}]_1^T, [\mathcal{Q}]_2^T, ..., [\mathcal{Q}]_{|\mathcal{Q}|}^T \end{bmatrix}^T, \tag{9}$$

where $[\mathcal{Q}]_i$ is the ith element of $\mathcal{Q}$. Thereafter, a formation-enforcing action that guides the team to the desired formation $\mathbf{q}_d$ is obtained by minimizing the quadratic error between the current relative poses and the desired relative poses

$$J(\mathbf{e}_F) = \frac{1}{2} \|\mathbf{e}_F\|^2, \tag{10}$$

$$\mathbf{e}_F = \varkappa_{\mathcal{G}}(\mathbf{q}_d) - \varkappa_{\mathcal{G}}(\mathbf{q}), \tag{11}$$

where $\|\cdot\|$ denotes the Euclidean norm.

To minimize the error described above, we apply the following local action which is expressed in the world coordinate frame

$$\begin{aligned} \dot{\mathbf{q}} &= -k_e \left(\frac{\partial J(\mathbf{e}_F)}{\partial \mathbf{q}} \right)^T \\ &= -k_e \left(\frac{\partial \varkappa_{\mathcal{G}}(\mathbf{q})}{\partial \mathbf{q}} \right)^T \left(\frac{\partial \mathbf{e}_F}{\partial \varkappa_{\mathcal{G}}} \right)^T \left(\frac{\partial J(\mathbf{e}_F)}{\partial \mathbf{e}_F} \right)^T \\ &= -k_e \left(\frac{\partial \varkappa_{\mathcal{G}}(\mathbf{q})}{\partial \mathbf{q}} \right)^T (-\mathbf{I})(\mathbf{e}_F) \\ &= k_e \left(\frac{\partial \varkappa_{\mathcal{G}}(\mathbf{q})}{\partial \mathbf{q}} \right)^T \mathbf{e}_F, \end{aligned} \tag{12}$$

where k_e is a constant gain.

If the observation graph $\mathcal{G}$ is connected and, as such, rigid in our case for a single agent i, the FEC can then be expressed as

$$\mathbf{u}_i = k_e \left(\sum_{j \in \mathcal{N}_i} c_{ij} \left(\mathbf{p}_{ij} - \mathbf{p}_{ij}^d \right) - \sum_{j \in \mathcal{N}_i} c_{ji} \mathbf{R}(\psi_{ji})^T \left(\mathbf{p}_{ji} - \mathbf{p}{ji}^d \right) \right),$$

$$\omega_i = k_e \left(-\sum_{j \in \mathcal{N}_i} c_{ij} \left(\mathbf{p}_{ij}^T \mathbf{S} \left(\mathbf{p}_{ij} - \mathbf{p}_{ij}^d \right) \right) \right. \tag{13}$$

$$\left. + \sum_{j \in \mathcal{N}_i} c_{ij} \left(\psi_{ij} - \psi_{ij}^d \right) - \sum_{j \in \mathcal{N}_i} c_{ji} \left(\psi_{ji} - \psi_{ji}^d \right) \right),$$

where $\mathcal{N}_i = \{1, 2, \cdots, N\} \setminus \{i\}$ and $\mathbf{S} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$. Details of constructing eq. (13) are clarified in appendix A. We express all the terms $\mathbf{p}_{ji}$ and ψ_{ji} as a function of $\mathbf{p}_{ij}$ and ψ_{ij}, which are available directly from the local sensors of agent i:

$$\mathbf{u}_i = k_e \left(\sum_{j \in \mathcal{N}_i} c_{ij} \left(\mathbf{p}_{ij} - \mathbf{p}_{ij}^d \right) \right.$$

$$\left. - \sum_{j \in \mathcal{N}_i} c_{ij} \mathbf{R}(\psi_{ij}) \left(\mathbf{R}(\psi_{ij})^T (-\mathbf{p}_{ij}) + \mathbf{R}(\psi_{ij}^d)^T \mathbf{p}_{ij}^d \right) \right), \tag{14}$$

$$\omega_i = k_e \left(\sum_{j \in \mathcal{N}_i} c_{ij} \left(\mathbf{p}_{ij}^{dT} \mathbf{S}^T \mathbf{p}_{ij} \right) + 2 \sum_{j \in \mathcal{N}_i} c_{ij} \left(\psi_{ij} - \psi_{ij}^d \right) \right).$$

Eq. (14) can be further simplified to

$$\mathbf{u}_i = k_e \left(\sum_{j \in \mathcal{N}_i} c_{ij} \left(\mathbf{p}_{ij} - \mathbf{p}_{ij}^d \right) + \sum_{j \in \mathcal{N}_i} c_{ij} \left(\mathbf{p}_{ij} - \mathbf{R}(\psi_{ij} - \psi_{ij}^d) \mathbf{p}_{ij}^d \right) \right),$$

$$\omega_i = k_e \left(\sum_{j \in \mathcal{N}_i} c_{ij} \left(\mathbf{p}_{ij}^{dT} \mathbf{S}^T \mathbf{p}_{ij} \right) + 2 \sum_{j \in \mathcal{N}_i} c_{ij} \left(\psi_{ij} - \psi_{ij}^d \right) \right). \tag{15}$$

In appendix B, the stability of this formation control is discussed and proven for an example case. However, we are not making claims of global stability for a general formation of arbitrarily connected agents; indeed, initial formations may exist that are within the region of attraction of a local minimum of the cost function $J(\mathbf{e}_F)$. For example, if local minimization of the orientation error balances out with the local minimization of position error. Such cases are typically rare, since a stable local minimum of the entire formation implies that there exists more than one state of $\varkappa_{\mathcal{G}}(\mathbf{q})$ such that, for all agents i, all elements of $[\mathbf{u}_i; \omega_i]^T$ as per (15) become zero simultaneously. In practice, initial formations are more

143

4

often unsuitable for use with FEC based on gradient descent, because such control does not take into account collision avoidance. This means that the agents may be controlled to *pass through* each other to achieve the desired formation, which can even be the case with a globally stable system.

We now replace $\mathbf{p}_{ij}$ and ψ_{ij} with their respective measurements that are contaminated with Gaussian noise $\mathbf{p}_{ij}^m \sim \mathcal{N}\left(\mathbf{p}_{ij}, \mathbf{C}_{ij}\right)$ and $\psi_{ij}^m \sim \mathcal{N}\left(\psi_{ij}, \sigma_{\psi ij}^2\right)$, where $\mathbf{C}_{ij}$ is the error covariance matrix for the relative position and σ_{ij}^2 is the variance for the heading angle difference. It is assumed that possible outliers were removed previously by the localization system itself. We then obtain the proportional control scheme to be applied by each agent

$$\mathbf{u}_i = k_e \left(\sum_{j \in \mathcal{N}_i} c_{ij} \overbrace{\left(\mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d\right)}^{\tau_{p_1}} + \sum_{j \in \mathcal{N}_i} c_{ij} \overbrace{\left(\mathbf{p}_{ij}^m - \mathbf{R}(\psi_{ij}^m - \psi_{ij}^d)\mathbf{p}_{ij}^d\right)}^{\tau_{p_2}} \right),$$

$$\omega_i = k_e \left(\sum_{j \in \mathcal{N}_i} c_{ij} \underbrace{\left(\mathbf{p}_{ij}^{dT} \mathbf{S}^T \mathbf{p}_{ij}^m\right)}_{\tau_{\psi_1}} + 2 \sum_{j \in \mathcal{N}_i} c_{ij} \underbrace{\left(\psi_{ij}^m - \psi_{ij}^d\right)}_{\tau_{\psi_2}} \right). \tag{16}$$

To ensure that the FEC is distributable, the directed observation graph must have no more than a single local sink, represented here by an observed agent that does not observe others. An agent is controlled based only on its own local observations without which it is passive. For multiple such agents, the desired relative pose between them will not be enforced. A single passive agent is permissible and in a specific application it may be used as the de-facto leader of a formation.

III. VECTOR COMPONENT ADJUSTMENT

In order to account for observation noise, we interpret the FEC input as a sum of vectors pointing from the desired relative pose of a neighbor to its actual measured relative pose. Examples of these components include the terms $\left(\mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d\right)$ and $\left(\psi_{ij}^m - \psi_{ij}^d\right)$, where only one operand is subject to noise with an assumed Gaussian distribution with covariance $\mathbf{C}_{ij}$ and standard deviation $\sigma_{\psi ij}$, respectively. A more complex case is represented by the term $\left(\mathbf{p}_{ij}^m - \mathbf{R}(\psi_{ij}^m - \psi_{ij}^d)\mathbf{p}_{ij}^d\right)$, with both operands burdened by observation noise.

The former case is depicted in Fig. 8. It includes an example of a 2D relative position measurement $\mathbf{p}_{ij}^m$ that has been measured by agent i, where the desired relative position of the neighbor is $\mathbf{p}_{ij}^d$. The depicted vector $\mathbf{d}_{ij}$ represents a translation of agent i that would unify the measured relative position $\mathbf{p}_{ij}^m$ of neighbor j and its desired relative position $\mathbf{p}_{ij}^d$. The measured relative position is burdened by observation noise with a Gaussian distribution, depicted as a cyan Gaussian blob.

If the system was continuous, then a normally distributed observation noise would not be an issue, since its effects would tend to cancel out before having any observable effect. However, in a real digital system such as the one discussed here, the sampling rate of relative-localization sensors is limited. This means that a given measurement error will affect the behavior of a formation for a non-infinitesimal time period,

which we need to suppress. The FEC is analyzed primarily as a discrete-time system, where each agent moves linearly based on the latest observation. The observations are delivered with constant rate, as this allows us to take into account the detrimental effects of sensory noise that scale with the sampling period.

A. Proportional control with noise

Let us analyze what will be the behavior of a simple single-agent system if it is subject to proportional control and a Gaussian observation noise. For the sake of simplicity, we will present this idea with a one-dimensional version of the control action represented by the term $\left(\mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d\right)$.

Consider an agent with a 1D state x with the dynamic model

$$x[k+1] = \Delta_{[k]} + x[k], \tag{17}$$

where k is the discrete time index and $\Delta_x[k]$ is the displacement at time instant k. Consider also a desired value d that the agent is trying to reach. In a noiseless scenario, the difference of d from $x[k]$ can be directly observed by the agent. A simple way to make the agent converge to the desired value is to use proportional control as seen in eq. (16). In a real robotic system measurements are retrieved with finite rate f. Therefore, we can set the control action at each iteration to be linear regression v based on the proportional factor k_e and on the latest relative measurement of d obtained at time index k. In this case, displacement will be set equal to

$$\Delta_{[k]} = \frac{v[k]}{f} = \frac{k_e}{f}\left(d - x[k]\right) = \frac{k_e}{f}\Delta_d = k_{ef}\Delta_d. \tag{18}$$

Since the dynamics of this system depend only on the ratio $\frac{k_e}{f}$ regardless of the individual scales of these two constants, we replace them with a single constant $k_{ef} = \frac{k_e}{f}$. Then, the control variable moves towards the desired value according to the geometric progression

$$x[k+1] = d + \left(x[k] - d\right)\left(1 - k_{ef}\right), \tag{19}$$

which, in this case, resolves to

$$x[k] = d + \left(x[0] - d\right)\left(1 - k_{ef}\right)^k. \tag{20}$$

In reality, the state of x changes in continuous time t in a piecewise linear motion of the form

$$\begin{aligned} k &= \lfloor tf \rfloor, \\ x(t) &= k_{ef}d\left(tf - k\right) \\ &+ \left(d + \left(x[0] - d\right)\left(1 - k_{ef}\right)^k\right)\left(1 - k_{ef}\left(tf - k\right)\right). \end{aligned} \tag{21}$$

However, the evolution of the state $x(t)$ can be approximated by the exponential descent function passing through the states $x[k]$, assuming that $k = tf$ as

$$x(t) = d + \left(x(0) - d\right)\left(1 - k_{ef}\right)^{tf}. \tag{22}$$

The controlled variable $x(t)$ will clearly converge to the desired value d if $k_{ef} \in (0, 2)$. The agent will converge monotonically if $k_{ef} \in (0, 1)$. If $k_{ef} \in (1, 2)$, the agent will converge to the desired value d, but it will be affected with damped oscillations passing over d in an alternating direction

5

at every iteration, which is contrary to our stated goals. We will therefore restrict the value of k_{ef} to $(0, 1)$.

Consider next that the agent only has a noisy measurement available of its current state x, denoted m:

$$m[k] \sim \mathcal{N}\left(x, \sigma_m[k]^2\right), \qquad (23)$$

or

$$
\begin{aligned}
m[k] &= x[k] + e_m[k], \\
e_m[k] &\sim \mathcal{N}\left(0, \sigma_m[k]^2\right),
\end{aligned}
\qquad (24)
$$

with $\sigma_m[k]$ being known. This means that although the relative position of the desired state $\Delta_d[k] = d - x[k]$ is also not exactly known, it is a sample from the known probability distribution

$$\Delta_d \sim \mathcal{N}\left(d - m[k], \sigma_m[k]^2\right) = \mathcal{N}\left(\Delta_m[k], \sigma_m[k]^2\right). \quad (25)$$

In other words, the probability density function of Δ_d is

$$p_{\Delta_d[k]}(s) = \frac{1}{\sigma_m[k]\sqrt{2\pi}} e^{-\frac{1}{2}\frac{s - \Delta_m}{\sigma_m[k]}^2}, \qquad (26)$$

as shown in Fig. 4. A simple maximum-likelihood approach is to select the setpoint s_{ml} of the state of the agent based on the measurement $m[k]$ as

$$
\begin{aligned}
s_{\mathrm{ml}}[k] &= m[k] + \mathbb{E}\left[\Delta_d[k]\right], \\
\mathbb{E}\left[\Delta_d[k]\right] &= d - m[k] = \Delta_m[k].
\end{aligned}
\qquad (27)
$$

Then the agent is controlled to reach the estimated relative position of the desired value, which is expected to maximize the likelihood of reaching d. We will therefore obtain per-iteration displacement of

$$\Delta_{[k]} = k_{ef}\left(\Delta_m[k]\right) = k_{ef}\left(d - m[k]\right). \qquad (28)$$

The state transition will be

$$
\begin{aligned}
x[k+1] &= \Delta_m[k] + x[k] = k_{ef}\left(d - m[k]\right) + x[k] \\
&= k_{ef}\left(d - x[k] - e_m[k]\right) + x[k] \\
&= d + \left(x[k] - d\right)\left(1 - k_{ef}\right) + k_{ef} e_m[k],
\end{aligned}
\qquad (29)
$$

which is similar to eq. (19) with the additional term of $k_{ef} e_m[k]$ representing the influence of sensory noise.

Lemma 3.1: Barring infinite measurement rate or a zero control action, it is *impossible* to use proportional control to enforce a stationary stable state if discrete measurements of this state used in the control law are subject to Gaussian noise.

Proof: If multiple agents follow the control law (without the modification proposed in this paper), the presented combination of proportional control and observation noise with Gaussian distribution actually leads to a Gaussian distribution of agents[1]. This is due to the fact that linear downscaling of values (represented by the multiplier $\left(1 - k_{ef}\right)$) about a common origin preserves the overall shape of their displacement, while only decreasing its scale. At the same time, repeatedly superposing Gaussian noise on a value (represented by the noise inherent in $k_{ef} e_m[k]$) leads to a behavior called random walk, the uncertainty of which also has Gaussian distribution. Although the distribution of the noise depends on the current

[1]In appendix C.2, we show that even with the proposed modification the resulting poses can be accurately approximated with a Gaussian distribution.

observation, in practice it typically changes continuously with most relative localization systems. At non-zero distances, the standard deviation is non-zero. Thus, to obtain an estimate of the stable-state error of the system, we will assume that the standard deviation of the measurement is $\sigma_{m,\mathrm{fin}}$, or the standard deviation associated with the desired distance. Then, the state in system (29) has a variance evolving according to

$$
\begin{aligned}
\mathrm{Var}\left[x[k+1]\right] &= \mathrm{Var}\left[k_{ef} e_m[k]\right] + \mathrm{Var}\left[(1 - k_{ef}) x[k]\right] \\
&= k_{ef}^2 \sigma_{m,\mathrm{fin}}^2 + (1 - k_{ef})^2 \mathrm{Var}\left[x[k]\right].
\end{aligned}
\qquad (30)
$$

Using the equation for geometric series, the closed form of the above is

$$
\begin{aligned}
\mathrm{Var}\left[x[k+1]\right] &= \frac{\sigma_{m,\mathrm{fin}}^2 k_{ef}\left(1 - \left(1 - k_{ef}\right)^{2k}\right)}{2 - k_{ef}} \\
&\quad + \left(1 - k_{ef}\right)^{2k} \mathrm{Var}\left[x[0]\right].
\end{aligned}
\qquad (31)
$$

As k approaches infinity, the variance of the system approaches

$$\mathrm{Var}\left[x[k \to \infty]\right] = \sigma_{ss}^2 = \frac{\sigma_{m,\mathrm{fin}}^2 k_{ef}}{2 - k_{ef}}. \qquad (32)$$

The two interacting behaviors in the system therefore converge to a Gaussian localization noise of individual agents with non-zero standard deviation of

$$\sigma_{ss} = \sigma_{m,\mathrm{fin}}\sqrt{\frac{k_{ef}}{2 - k_{ef}}}. \qquad (33)$$

∎

This is the negative effect that is mitigated using the proposed approach.

B. 1D restraining

In order to combat the oscillations while still keeping the required agility of the multi-robot system, we propose to exploit the sensory model as follows. Consider that if the error $\Delta_m[k]$ is large compared to $\sigma_m[k]$ then $\Delta_m[k]$ consists predominantly from the true error $\Delta_d[k]$ with the additive measurement noise being a minor component. On the other hand, if the agent is measured to be close to d and $\Delta_m[k]$ becomes small compared to $\sigma_m[k]$, the measured error becomes likely to stem predominantly from observation noise. In other words, the closer the agent measures itself to be to the desired state d, the more likely it is that correcting the error Δ_m is unnecessary and will increase Δ_d by retreating or by overshooting the target. If this is not taken into account, an agent near d will tend to frequently change its direction of motion, which is contrary to our goals. Thus, if $\Delta_m[k]$ is large compared to $\sigma_m[k]$ then the measured difference vector can be safely applied in a proportional control for rapid convergence, while if $\Delta_m[k]$ is small w.r.t. $\sigma_m[k]$ then the control action should be suppressed.

In order to derive a modified action satisfying these requirements, we first propose an abstract control scheme with a region of passivity around the desired state. As before, the system has dynamics of the form

$$x[k+1] = \Delta_{\mathrm{res}[k]} + x[k] = \frac{v_{x\mathrm{res}}[k]}{f} + x[k]. \qquad (34)$$
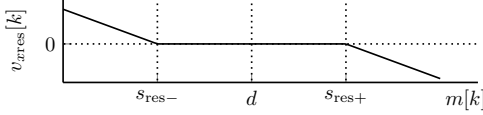
Fig. 3: The velocity set with our proposed controller in 1D case. The region within $m[k] \in [s_{\text{res}-}, s_{\text{res}+}]$ represents a region where the agent remains purposefully passive.

Here, $v_{x\text{res}}$ is a controllable velocity defined by

$$v_{x\text{res}}[k] = k_e \left(\text{clamp}\left(s_{\text{res}}\left(\Delta_m\right) - m[k], \Delta_m[k]\right)\right), \quad (35)$$

where $s_{\text{res}}\left(\Delta_m[k]\right)$ is a position with constant offset $|o_{\text{res}}|$ from d towards m, s.t. $o_{\text{res}} \in \mathbb{R}_{<0}$,

$$s_{\text{res}} = d + \text{sign}\left(\Delta_m[k]\right)o_{\text{res}} = \begin{cases} s_{\text{res}-} & \text{if} \quad \Delta_m[k] > 0 \\ s_{\text{res}+} & \text{otherwise.} \end{cases} \quad (36)$$

The function

$$\text{clamp}\,(y, a) = \begin{cases} y & \text{if} \quad y \cdot a \in (0, a^2] \\ 0 & \text{otherwise,} \end{cases} \quad (37)$$

nullifies the control action within the region $[s_{\text{res}-}, s_{\text{res}+}]$. Thus, $v_{x\text{res}}[k]$ is a piecewise-linear function of $m[k]$, as shown in Fig. 3. This control will steer an agent towards a specific *restrained setpoint* s_{res} offset from the desired state d, with a velocity decreasing to zero. This way the agent will not stop abruptly at the setpoint.

However, due to the influence of sensory noise on the measurement $m[k]$, the control will have a stochastic behavior. The agent may sometimes incorrectly evaluate itself as being inside the region $[s_{\text{res}-}, s_{\text{res}+}]$. Conversely, the agent may evaluate itself as being outside said region when $|d - x| < o_{\text{res}}$. This is why the controlled velocity (35) has a dead zone around the desired state. If s_{res} is set up correctly, it will prevent the agent from reacting to measurements that are statistically likely to cause *overshooting* of the desired state d by the agent.

To take into account the distribution of sensor noise, we need to condition the restrained setpoint s_{res} on the standard deviation $\sigma_m[k]$ of the current measurement $m[k]$. Though using a dead zone for control is intuitive in and of itself, it is the specific design of this dead zone that we consider to be a contribution of this paper. We will detail how the setpoint s_{res} should be set in the following section.

1) Setpoint selection: Let us define *overshooting* of the controller as

$$\text{sign}\,(s[k] - d) \neq \text{sign}(x[k] - d), \quad (38)$$

where $s[k]$ is a future state to be reached by the agent using a given control - a *setpoint*. The probability of overshooting d by reaching setpoint s is

$$P_{\text{over}} = P\left(\text{sign}\,(s[k] - d) \neq \text{sign}\,(x[k] - d)\right). \quad (39)$$

The maximum-likelihood approach in section III-A uses a setpoint s_{ml} based on the current mean of the measured relative position of d. Since this measurement is subject to Gaussian noise with the standard deviation $\sigma_m[k]$, the controller therein
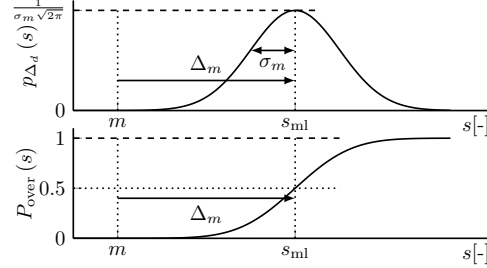
Fig. 4: A simple case of 1D Gaussian distribution. The example is unitless, as this consideration applies to any measurement burdened by Gaussian noise. The true difference Δ_d of the desired state d w.r.t. the current state x is unknown, since we only have a noisy measurement m of x. However, we do assume the *distribution* of the noise present in m to be known in the from of a Gaussian distribution with standard deviation $\sigma_m[k]$ centered on x. Conversely, the available measurement of Δ_d is represented by a Gaussian distribution with mean $\Delta_m[k] = d - m[k]$ and standard deviation $\sigma_m[k]$. The probability P_{over} of *overshooting* the target d depends on the setpoint s chosen, and is equal to the CDF of the Gaussian distribution of Δ_d. If we merely choose $s_{\text{ml}} = m[k] + \Delta_m[k]$, then we risk overshooting the target with non-negligible probability of 0.5.

risks overshooting the actual desired value d with a probability

$$\begin{aligned} P_{\text{over,ml}} &= \Phi\left(\frac{s_{\text{ml}}[k] - d}{\sigma_m[k]}\right) \\ &= \Phi\left(\frac{m[k] + \Delta_m[k] - d}{\sigma_m[k]}\right) \\ &= \Phi\left(\frac{m[k] + (d - m[k]) - d}{\sigma_m[k]}\right) \\ &= \Phi\,(0) = 0.5, \end{aligned} \quad (40)$$

where $\Phi(\cdot)$ is the cumulative probability density function of the Gaussian distribution with zero mean and unit standard deviation, as illustrated in Fig. 4. Note that, the probability of 0.5 is fairly high and thus the maximum-likelihood approach will be switching the direction of $v[k]$ often. Combined with the finite sampling frequency f, this causes stochastic oscillations of the agent about the desired state d that scale with k_{ef}, as per eq. (33).

Instead of using the maximum-likelihood approach, we can define a *restrained setpoint* (Fig. 5) based on a specified maximum *allowed* probability $\ell \in (0, 0.5]$ of overshooting, as defined above. Due to sensory noise having a known statistical distribution, with mean $m = m[k]$ and $\sigma_m = \sigma_m[k]$, a s_{res} conforming to this probability ℓ is

$$\begin{aligned} s_{\text{res}}\,(m, \sigma_m) &= d + \text{sign}\,(\Delta_m)\sigma_m \Phi^{-1}(\ell) \\ &= m + \Delta_m + \text{sign}\,(\Delta_m)\sigma_m \Phi^{-1}(\ell), \end{aligned} \quad (41)$$

where $\Phi^{-1}(\cdot)$ is the inverse function of the cumulative probability function of Gaussian distribution with zero mean and unit standard deviation. We will use s_{res} in the proposed control in eq. (34), which makes the state change between subsequent measurements become

$$\begin{aligned} \Delta_{[k]\text{res}} &= \frac{v_{x\text{res}}[k]}{f} = \frac{k_e}{f}\,\text{clamp}\,(s_{\text{res}} - m[k], \Delta_m[k]) \\ &= k_{ef}\,\text{clamp}\,\left(\text{sign}\,(\Delta_m[k])\sigma_m[k]\Phi^{-1}\,(\ell) + d, \Delta_m\right). \end{aligned} \quad (42)$$

This makes for a flexible control that is robust to sensory noise, without the need for manually tuning the proportional factor with regards to the specific standard deviation of the

7
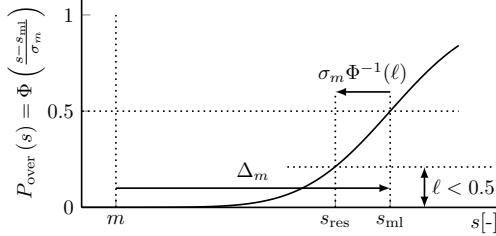
Fig. 5: If the measured relative value of a target state is subject to observation noise, controlling the current state of the system towards the mean of that measurement at s_{ml} risks overshooting the target with probability $P_{\mathrm{over}} = 0.5$. This probability can be reduced to an arbitrary value $\ell < 0.5$ by choosing a restrained setpoint s_{res} that is closer to the current state than s_{ml} by $\left| \sigma_m[k]\Phi^{-1}(\ell) \right|$. If $\ell < 0.5$, then $\sigma_m[k]\Phi^{-1}(\ell) < 0$.

sensor noise present. The user only needs to set the probability ℓ. The control will steer the agent to a state defined by the limit on the probability of overshooting the target, and if it finds the agent to be located beyond this threshold it will become inactive. In essence, this acts as a spatial probabilistic motion filter that allows agents to get closer to the desired state than s_{res} through the effects of the sensory noise, but it will not actively steer it there if the probability of overshooting is too high. The probability that the agent will actively move in a given iteration decreases with $|d - x|$ according to eq. (87) in the appendix C.1. Therein, we show that the minimum motion probability is 2ℓ at $x = d$. Therefore the state of the agent is more likely to remain near the desired state, in essence narrowing the probability distribution of the state of the agent towards the desired value in comparison with pure proportional control. The system is statistically stable overall, as is shown in appendix C.1, and the expected state of the agent converges towards d while the expected variance at the stable state is

$$\sigma_{ss,\mathrm{res}}^2 = \frac{k_{ef}\sigma_{m,\mathrm{fin}}^2}{2 - k_{ef}} \exp\left(\beta(k_{ef})\Phi^{-1}(\ell) \right) \qquad (43)$$

where $\beta(\cdot)$ is an empirical function with example values found in table II. Reducing ℓ therefore reduces the stable-state noise in the system. For example, if $k_{ef} = 0.5$ and $\ell = 0.3$ the agents will be displaced from the target in the stable state by zero-mean amount with variance

$$\sigma_{ss,\mathrm{res}} = 0.8051\sigma_{ss}. \qquad (44)$$

If ℓ is set to 0.5, the control will be equal to the naive proportional control in III-A, as in the case $s_{\mathrm{res}} = d$, with the dead zone collapsed into a point. Indeed, if $|d - x| \gg \sigma_m$, the agent will behave similarly to being driven by a proportional control, which means that the rate of convergence to the desired formation is not significantly reduced.

The net effect of the above is that the *stable state* variance of the agent's displacement from d is *reduced* compared to σ_{ss} from eq. (33). The noise reduction is dependent on ℓ and on k_{ef}, and the specifics of the behavior are analyzed in depth in appendix C.1.

2) Analysis of the control performance: To first demonstrate the operation of the proposed control scheme in a simulated 1D system, we establish a large number (10,000) of virtual agents with one degree of freedom each, as these agents operate on a scalar variable x. Using a large number

of agents in the case of the presented simulation allows us to observe statistical properties of the proposed control. Without loss of generality, we can choose the unit of x to be meters, so that we may provide an intuitive understanding in terms of robotic motion control with the position being the controlled variable. The parameters of the presented simulation are such that the relevant effects of the proposed control can easily be demonstrated on reasonably scaled plots. However, the presented effects persist across a wide range of parameters, albeit with changes in their relative magnitudes. The agents were first pseudo-randomly displaced according to a Gaussian distribution around the mean equal to d, with the standard deviation of their positions being $\sigma_I = 100\,\mathrm{m}$. Each agent then measured its distance to the desired value at a rate $f = 10\,\mathrm{Hz}$, where each measurement is burdened by pseudo-random additive noise with a Gaussian distribution of $\sigma_m = 3\,\mathrm{m}$. The whole simulation was executed for a descending range of thresholds ℓ, starting with $\ell = 0.5$. If $\ell = 0.5$, the control is equivalent to the proportional control without the proposed modification,which is supported by the simulation where the estimated σ_t, obtained as the standard deviation of the agent positions from their mean at a given time step, converged to the theoretical value from eq. (33) of $\sigma_{ss} = 1.73\,\mathrm{m}$ for the selected parameters at $\ell = 0.5$. This represents the upper bound of the stable-state localization noise in the system. Fig. 6 shows the time evolution of three relevant variables in the simulation: the average difference $\overline{\Delta v}(t)$ in rate of change in d between two consecutive measurements, the average distance $\overline{\Delta_d}(t)$ from the desired position d, and the standard deviation $\sigma_a(t)$ of the distribution of the agents. By only decreasing the threshold ℓ, the rate of change in x becomes progressively more steady, which is exactly the goal we set out to achieve. In addition, although the coalescence of the agents towards a stable distribution gets slower with decreasing ℓ, the standard deviation of their distribution *actually becomes smaller*. This means that, despite the fact that we are not controlling them directly towards their measured relative position Δ_m, the agents end up closer to the desired state d than they would have otherwise. Lastly, we analyzed how the proposed system compares to a pure proportional control in terms of the tradeoff between the speed of convergence and the stable-state distribution and velocity. To do this, we have performed multiple simulations of the above system for a set of values of ℓ, including $\ell = 0.5$ representing the behavior of pure proportional control. We have swept through values of $k_{ef} \in [0.02, 0.97]$ to show that merely tuning the proportional control does not provide better results than using the proposed *restraining* technique. For each run of the simulation, we have evaluated three parameters:

- Time t_c to convergence to stable-state oscillations
- The standard deviation σ_t from the desired state in the stable-state
- Average change $\overline{\Delta v}$ in velocity between subsequent linear movements based on their latest measurements

To evaluate these parameters, we will define the time of convergence to the stable state in the simulation data with M samples as the first instance when the state becomes closer

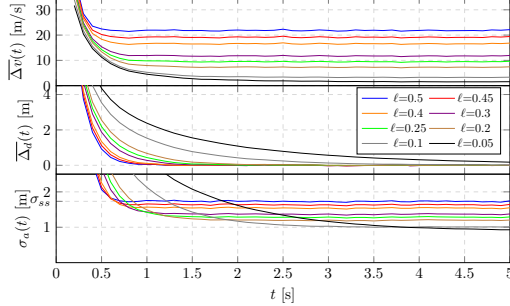CHAPTER 4. RESEARCH MOTIVATED BY THE *UVDAR* SYSTEM

Fig. 6: Simulation results of the 1D case of proportional control with the restrained target value. Each plot is obtained by averaging the motion of 10,000 virtual agents with pseudo-random initial distribution and observation noise. For reproducibility, the initial conditions were the same for each value of ℓ. From top, these time plots represent the average change in velocity between actions from two subsequent target measurements, the average distance from the target position, and the standard deviation from the target. The parameter values for these plots were $k_{ef} = 0.5$, $\sigma_m = 3$ m and $f = 10$ Hz. The agents started in a Gaussian distribution centered on the target with a standard deviation of 100 m.

to the desired state than 3 times the standard deviation of the rest of the recorded states, or:

$$t_c = \frac{k_c}{f},$$

$$k_c = \min \left\{ k \in 0..M : \left| x[k-1] - d \right| > 3\sigma_k \right\}, \quad (45)$$

$$\sigma_{\text{fin}}[k] = \sqrt{\text{var}\left(\left\{ x[k'] - d : k' \in k..M \right\} \right)}.$$

The stable-state standard deviation σ_t and average convergence velocity $\overline{\Delta v}$ are then defined as

$$\sigma_t = \sigma_{\text{fin}}[k_c], \quad (46)$$

$$\overline{\Delta v} = \text{mean}\left(\left\{ (x[k] - x[k-1])\, f : k \in 1..M \right\} \right). \quad (47)$$

Using the above definitions, we have produced Fig. 7, where each point in the plots was averaged from 1,000 runs with the same parameters. The figure shows that using the proposed technique can obtain better combinations of behavior properties than what is possible by merely changing the value of k_{ef}. This is evident by all the plots with $\ell < 0.5$ contained *under* the plot of $\ell = 0.5$ corresponding to the pure proportional control. We arbitrarily chose $\sigma_m = 1$ and $f = 10$ Hz, since these variables merely scale the values of σ_t and $\overline{\Delta v}$ across the board.

The lower the value of k_{ef}, the lower the improvement achieved by using restrained target values when compared to using the measured values directly. Lowering the k_{ef} by reducing k_e diminishes the influence of measurement noise on the control action, although doing so also significantly slows down convergence towards the desired values.

It must be noted that, in a general formation control, agents strive to reach a relative displacement with respect to other agents. If the observed agent is static, the observing agent behaves as detailed above, where we substitute the desired state d with the desired displacement with respect to the other agent. However, two agents may observe each other at the same time. In case of mutual observation, the agents will both move towards their desired relative displacement, each
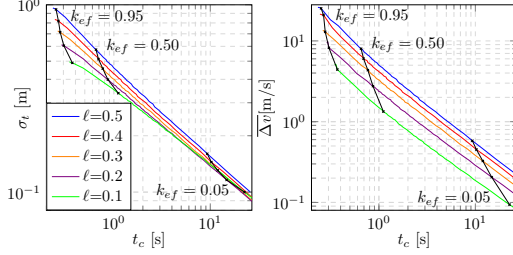
Fig. 7: On the left, we show the comparison of the trade-off between stable-state standard deviation σ_t from the desired state and the time t_c it took to reach the stable-state oscillation threshold for various values of ℓ. The curves show the presented value w.r.t. a sweep across values of $k_{ef} \in [0.02, 0.97]$. On the right, we show a similar comparison between the average change in velocity between subsequent measurement-induced motions and the time of convergence, in the same ranges of k_{ef}. These plots are produced by 1D simulation of the presented dynamic system, with framerate $f = 10$ and $\sigma_m = 1$ for a specific scale. The plot for $\ell = 0.5$ corresponds to pure proportional control; in either case, we obtain better combination of parameters by applying our restraining technique, since the other plots are contained under the curve of $\ell = 0.5$.

according to their current observation. In appendix D, we show that in such a case the range of k_{ef} for which the system is generally stable reduces from $(0, 2)$ to $(0, 1)$, which in turn translates to the range $(0, 0.5)$ for monotonical convergence to the desired displacement. This is not a stringent requirement, since $k_{ef} = 0.5$ is in practice an extreme value - for example, with $f = 10$ Hz this translates to a local proportional factor $k_e = 5$ times the displacement. The observation does imply however, that the increasing number of connections in the formation leads to increased limits on the dynamics that enable convergence in a controlled manner. As we note in the appendix, the dynamics of the mutual-observation case will eventually turn into the behavior of a single-observation case, since one of the two agents will stop with high probability near the desired displacement, after which the remaining agent will move as if with respect a static desired state.

C. Extension to $\mathbb{R}^3 \times \mathbf{S}^1$

Returning to the FEC derived from eq. (16), we will now explore how the *restraining* technique above can be exploited for a more complex, multi-dimensional problem.

We propose to mitigate the oscillations introduced in the controller of proportional control (16) due to the noisy measurements. To do this, we will apply the restraining techniques developed in section III-B, particularly eqs. (35) and (41). However, these equations are developed for a one-dimensional system while (16) is multi-dimensional. Thus, before we apply (35) and (41) to (16), we will perform some projections to transform (41) into a one-dimensional function. For each term in (16) (i.e., τ_{p_1}, τ_{p_2}, τ_{ψ_1}, and τ_{ψ_4}), we will perform the following. Through projections, we will reduce the dimensionality to 1, determine the corresponding setpoint (41), and provide the control velocity based on (35).

Let us next extend the overshooting definition provided for the one dimensional case in (38) to generalize the higher dimensional case. We define overshooting as a change from the original state $\mathbf{p}_{ij}[k]$ to a new state $\mathbf{p}_{ij}[k+1]$, where its orthogonal projection onto the line connecting $\mathbf{p}_{ij}[k]$ and the desired state $\mathbf{p}_{ij}^d$ falls beyond the desired target from the

Fig. 8: Simple case of a 2D relative position measurement with Gaussian noise.

perspective of the original position, i.e. *if*:

$$\langle \mathbf{p}_{ij}[k+1] - \mathbf{p}_{ij}[k], \mathbf{p}_{ij}^d - \mathbf{p}_{ij}[k]\rangle > \left\| \mathbf{p}_{ij}^d - \mathbf{p}_{ij}[k] \right\|^2 . \quad (48)$$

where $\langle \cdot, \cdot \rangle$ is the inner product. Additionally, we move the *desired* relative poses of neighbors towards their associated *measured* poses. This is reversed w.r.t. section III-B, since it would be counter-intuitive in the simple 1D case presented therein. The *desired* relative poses of each agent's neighbors are attached to the body frame of said agent, while the measured poses are located independently in the surroundings.

In cases of bilateral observation, note that two mutually-observing agents will both follow the proposed control. In appendix D we show in a 1D case that this behavior will transform into unilateral observation once one of the agents enters its own dead zone.

Applying the dead zone directly to the original cost function is not practical in our case, primarily due to the fact that the resulting FEC would still depend on ground-truth states of the agents due to the geometrical interdependence of the relative position and orientation of agents. We apply the proposed restraining technique *post-hoc* directly on the FEC as formulated for an ideal system, as this allows us to address the effects of noise on every occurring instance of a noisy measurement.

D. Term τ_{p_1}

To see how this will make our multidimensional problem compatible with our 1D restraining approach, consider that the measurement has multivariate Gaussian distribution with a known covariance $\mathbf{C}_{ij}$ and mean $\mathbf{p}_{ij}$

$$\mathbf{p}_{ij}^m[k] \sim \mathcal{N}\left(\mathbf{p}_{ij}, \mathbf{C}_{ij}\right), \quad (49)$$

and thus, conversely, given the measurement $\mathbf{p}_{ij}^m[k]$, the likelihood that $\mathbf{p}_{ij}$ is located at the point $\mathbf{p}$ is

$$P\left(\mathbf{p}_{ij} = \mathbf{p}\right) = \frac{1}{\sqrt{(2\pi)^k |\mathbf{C}_{ij}|}} e^{\frac{1}{2}\left(\mathbf{p} - \mathbf{p}_{ij}^m[k]\right)^T \mathbf{C}_{ij}^{-1}\left(\mathbf{p} - \mathbf{p}_{ij}^m[k]\right)}. \quad (50)$$

If we restrict the point $\mathbf{p}$ to lie along the line defined by the points $\mathbf{p}_{ij}^d$ and $\mathbf{p}_{ij}^m[k]$, then the likelihood function becomes a 1D Gaussian distribution with mean $\mathbf{p}_{ij}^m[k]$ and variance σ_{pij}^2.

To apply the eq. (41), we need to derive the expression for the standard deviation σ_{pij} of the 1D Gaussian distribution described above. We do this by calculating the Mahalanobis distance between $\mathbf{p}_{ij}^m[k]$ and $\mathbf{p}_{ij}^d$, and dividing their Euclidean distance by it. Thus, we obtain

$$\sigma_{pij} = \frac{\left\| \mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d \right\|}{\sqrt{\left(\mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d\right)^T \mathbf{C}_{ij}^{-1}\left(\mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d\right)}}. \quad (51)$$
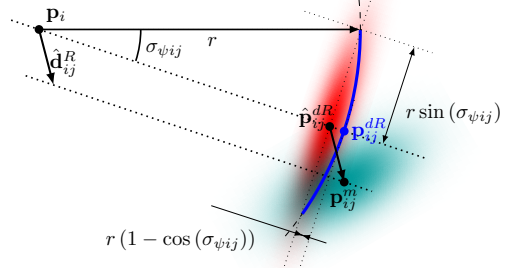
Fig. 9: Extended case of 2D relative pose measurement with both ends of the difference vector being burdened by observation noise.

Consequently, we can express the setpoint $\mathbf{s}_{p_1}$ as

$$\begin{aligned}
\mathbf{s}_{p_1} &= \sigma_{pij} \frac{\mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d}{\left\| \mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d \right\|} \left(\Phi^{-1}\left(\ell\right)\right) + \mathbf{p}_{ij}^m \\
&= \frac{\mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d}{\sqrt{\left(\mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d\right)^T \mathbf{C}_{ij}^{-1}\left(\mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d\right)}} \Phi^{-1}\left(\ell\right) + \mathbf{p}_{ij}^m.
\end{aligned} \quad (52)$$

It is with respect to this target position that we derive a partial positional action that asymptotically approaches a maximum admissible probability ℓ of overshooting our original desired relative position.

E. Term τ_{p_2}

The term τ_{p_2} is composed of two parts: a three dimensional Gaussian random variable (i.e. $\mathbf{p}_{ij}^m$) and three dimensional non-Gaussian random variable (i.e. $\mathbf{R}(\psi_{ij}^m - \psi_{ij}^d)\mathbf{p}_{ij}^d$). The non-Gaussian variable is the result of a three-dimensional nonlinear function applied to a single Gaussian variable (ψ_{ij}^m). We will hereby refer to this non-Gaussian random variable as $\mathbf{p}_{ij}^{dR}$. This is illustrated in Fig. 9, where the noise of $\mathbf{p}ij^m$ is illustrated as a cyan Gaussian blob and one standard deviation of the probability distribution of $\mathbf{p}_{ij}^{dR}$ is shown in blue.

We start by approximating $\mathbf{p}_{ij}^{dR}$ with a new Gaussian variable $\hat{\mathbf{p}}_{ij}^{dR}$, such that one standard deviation of the distribution of $\hat{\mathbf{p}}_{ij}^{dR}$ encloses one standard deviation of the distribution of $\mathbf{p}_{ij}^{dR}$ (Fig. 9). This method differs from the more standard method of using the first term of the Jacobian, which would not take into account the true distribution being biased in the direction of $-\mathbf{p}_{ij}^{dR}$. We obtain the mean $\hat{\mathbf{p}}_{ij}^{dR}$ according to

$$\hat{\mathbf{p}}_{ij}^{dR} = \left[\left[[\mathbf{p}_{ij}^{dR}]_1 \quad [\mathbf{p}_{ij}^{dR}]_2 \right] \cos\left(\sigma_{\psi ij}\right) \quad [\mathbf{p}_{ij}^{dR}]_3 \right]^T, \quad (53)$$

where $[\cdot]_n$ denotes the n-th element of a vector.

We then generate a Gaussian distribution centered on this new starting point, such that its eigenvalue corresponding to the vertical eigenvector is zero, making this a degenerate or "flat" distribution. Together with the other two eigenvalues corresponding to the eigenvectors pointing tangentially and perpendicularly to horizontal components of $\hat{\mathbf{p}}^{dR}$, respectively,

CHAPTER 4. RESEARCH MOTIVATED BY THE *UVDAR* SYSTEM

the covariance $\mathbf{C}_{tij}$ of the new distribution is constructed as:

$$\mathbf{v}_{ij1}^d = \begin{bmatrix} [\mathbf{p}_{ij}^{dR}]_1 & [\mathbf{p}_{ij}^{dR}]_2 & 0 \end{bmatrix}^T,$$

$$\mathbf{v}_{ij2}^d = \begin{bmatrix} -[\mathbf{p}_{ij}^{dR}]_2 & [\mathbf{p}_{ij}^{dR}]_1 & 0 \end{bmatrix}^T,$$

$$\mathbf{v}_{ij3}^d = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T,$$

$$\mathbf{V}_{ij} = \begin{bmatrix} \dfrac{\mathbf{v}_{ij1}^d}{\left\| \mathbf{v}_{ij1}^d \right\|} & \dfrac{\mathbf{v}_{ij2}^d}{\left\| \mathbf{v}_{ij2}^d \right\|} & \dfrac{\mathbf{v}_{ij3}^d}{\left\| \mathbf{v}_{ij3}^d \right\|} \end{bmatrix}, \quad (54)$$

$$\sigma_{\psi ij}^c = \min\left(\sigma_{\psi ij}, \frac{\pi}{2} \right),$$

$$\mathbf{L}_{ij} = \mathrm{diag}\left(\left\| \mathbf{v}_{ij1}^d \right\| \left[1 - \cos\left(\sigma_{\psi ij}^c\right)^2 \sin\left(\sigma_{\psi ij}^c\right)^2 \delta^2 \right] \right),$$

$$\mathbf{C}_{tij} = \mathbf{V}_{ij}\mathbf{L}_{ij}\mathbf{V}_{ij}^{-1},$$

where δ is an arbitrarily small, non-zero value chosen to prevent numerical instabilities. This new distribution is shown in red in Fig. 9 and indicates that the starting point can also lie *inside* the circle containing the original distribution, which is a conservative assumption.

Thus, we approximate the term τ_{p_2} with the random Gaussian variable $\mathbf{p}_{ij}^m - \hat{\mathbf{p}}_{ij}^{dR}$ with covariance matrix:

$$\mathbf{C}_{cij} = \mathbf{C}_{ij} + \mathbf{C}_{tij}. \quad (55)$$

Then the set point for this term is

$$\mathbf{s}_{p_2} = \frac{\mathbf{p}_{ij}^m - \hat{\mathbf{p}}_{ij}^{dR}}{\sqrt{\left(\mathbf{p}_{ij}^m - \hat{\mathbf{p}}_{ij}^{dR}\right)^T \mathbf{C}_{cij}^{-1} \left(\mathbf{p}_{ij}^m - \hat{\mathbf{p}}_{ij}^{dR}\right)}} \Phi^{-1}\left(\ell\right) + \mathbf{p}_{ij}^m. \quad (56)$$

F. Term τ_{ψ_1}

Let us start by factorizing this term as follows:

$$\left(\mathbf{p}_{ij}^d\right)^T \mathbf{S}^T \mathbf{p}_{ij}^m = \left\| \mathbf{F}\mathbf{p}_{ij}^d \right\| \cdot \left\| \mathbf{F}\mathbf{p}_{ij}^m \right\| \cdot \sin\left(\alpha\right), \quad (57)$$

where α is the angle formed by the desired and measured bearings of agent j from the local perspective of agent i, with the positive orientation being the same as the positive orientation of the angles ψ, and with $\mathbf{F} = \mathrm{diag}\left(\begin{bmatrix} 1 & 1 & 0 \end{bmatrix}^T \right)$ removing the vertical components from the two vectors.

Here, we once again chose a non-standard approach instead of directly calculating the standard deviation of this term by multiplying the differentiation of the term based on $\mathbf{p}_{ij}^m$ with the covariance $\mathbf{C}_{ij}$. This was done in order to avoid the undue scaling of the resulting standard deviation with $\mathbf{p}_{ij}^d$, which would otherwise occur as a side-effect from the prior linearization used to obtain the local control eq. (16).

Since a change in the orientation of the observing agent does not affect the norm of $\mathbf{p}_{ij}^m$, the *restraining* applied will depend on $\sin\left(\alpha\right)$. We will evaluate the Gaussian distribution function at α. For this purpose, we need to know the standard deviation $\sigma_{\beta ij}$ of the bearing measurement $\beta_{ij} = \mathrm{atan2}\left(\mathbf{p}_{ij}^m(2), \mathbf{p}_{ij}^m(1)\right)$. This value is related to the relative position observation noise, in the sense that the true
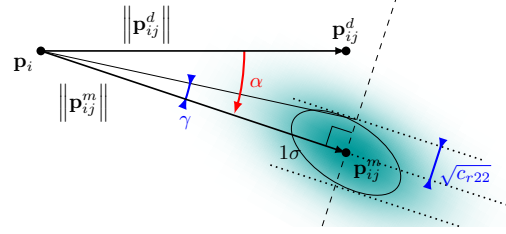
Fig. 10: Special case of 2D relative pose measurement for the orientation command based on the relative bearing error. The ellipse denotes the probability density corresponding to one standard deviation.

distribution of the bearing measurement noise corresponds with the projection of the distribution of the relative position measurement noise onto a circle centered over the observer, conditioned on the bearing angle. The exact calculation of such a projection in the general case is complicated and non-linear. However, for the majority of the practical scenarios the distances between the observed agents and the observers are reasonably large with respect to the components of the covariance matrix $\mathbf{C}_{ij}$. We can thus approximate the projection by the following orthogonal projection:

$$\mathbf{C}_r = \mathbf{R}(-\beta_{ij})\mathbf{C}_{ij}\mathbf{R}(-\beta_{ij})^T = \begin{bmatrix} \begin{smallmatrix} c_{r11} & c_{r12} \\ c_{r21} & c_{r22} \end{smallmatrix} & \\ & \ddots \end{bmatrix},$$

$$\gamma = \arctan\left(\frac{\sqrt{c_{r22}}}{\left\| \mathbf{p}_{ij}^m \right\|} \right). \quad (58)$$

This geometric consideration is illustrated in Fig. 10. Thus, for our 3D case an approximation $\sigma_{\beta ij}'$ of the standard deviation $\sigma_{\beta ij}$ of the above distribution can be obtained as

$$\sigma_{\beta ij} \approx \gamma \approx \tan\left(\gamma\right),$$

$$\sigma_{\beta ij}' = \frac{\sqrt{c_{r22}}}{\left\| \mathbf{p}_{ij}^m \right\|}. \quad (59)$$

Next, we modify the proportional action corresponding to $\left(\mathbf{p}_{ij}^{dT}\mathbf{S}^T\mathbf{p}_{ij}^m\right)$ for an equivalent effect, as with eq. (52), (62), and (56). We obtain a restrained modification of $\mathbf{p}_{ij}^m$ by rotating it in a horizontal plane towards the horizontal projection of $\mathbf{p}_{ij}^d$, in the sense of the shortest angle. Therefore,

$$\hat{\mathbf{p}}_{ij}^{c3} = \mathbf{R}\left(\sigma_{\beta ij}' \, \mathrm{sign}\left(\zeta\left(\mathbf{p}_{ij}^d\right) - \zeta\left(\mathbf{p}_{ij}^m\right) \right) \Phi^{-1}\left(\ell\right) \right) \mathbf{p}_{ij}^m, \quad (60)$$

where $\zeta\left(\cdot\right)$ calculates the horizontal bearing angle of a vector in the argument as $\zeta\left(\mathbf{x}\right) = \mathrm{atan2}\left([\mathbf{x}]_2, [\mathbf{x}]_1\right)$.

Applying this to eq. (57) yields the modified term:

$$\left(\mathbf{p}_{ij}^{dT}\mathbf{S}^T\hat{\mathbf{p}}_{ij}^{c3}\right) = \left\| \mathbf{p}_{ij}^d \right\| \cdot \left\| \mathbf{p}_{ij}^m \right\| \cdot \sin\left(\alpha - \mathrm{sign}\left(\alpha\right) \sigma_{\beta ij}' \Phi^{-1}\left(\ell\right) \right). \quad (61)$$

Since the term approximately scales with the square of the neighbor distance, care should be taken in order to prevent divergent rotational behavior by ensuring that the rotational action it induces when multiplied by $k_{ef} = \frac{k_e}{f}$ does not exceed $\pm \pi \mathrm{rad}$.

G. Term τ_{ψ_2}

In this case, the term is one-dimensional Gaussian already, so we directly apply the eq. (41):

$$s_{\psi_2} = \sigma_{\psi ij} \, \text{sign} \left(\psi_{ij}^m - \psi_{ij}^d \right) \Phi^{-1} \left(\ell \right) + \psi_{ij}^m. \tag{62}$$

H. Combining terms

We now have a completely modified variant of eq. (16) that reduces changes in velocity induced by measurement noise, thus mitigating oscillations:

$$\begin{aligned}
\mathbf{u}_i = & \, k_e \sum_{\substack{j=1,\ldots,N \\ c_{ij}=1}} \text{clamp} \left(\mathbf{s}_{p_1} - \mathbf{p}_{ij}^d, \mathbf{p}_{ij}^m - \mathbf{p}_{ij}^d \right) \\
& + k_e \sum_{\substack{j=1,\ldots,N \\ c_{ij}=1}} \text{clamp} \left(\mathbf{s}_{p_2} - \hat{\mathbf{p}}_{ij}^{dR}, \mathbf{p}_{ij}^m - \hat{\mathbf{p}}_{ij}^{dR} \right) \\
\omega_i = & \, k_e \sum_{\substack{j=1,\ldots,N \\ c_{ij}=1}} \text{clamp} \left(\mathbf{p}_{ij}^{dT} \mathbf{S}^T \hat{\mathbf{p}}_{ij}^{c3}, \mathbf{p}_{ij}^{dT} \mathbf{S}^T \mathbf{p}_{ij}^m \right) \\
& + 2 k_e \sum_{\substack{j=1,\ldots,N \\ c_{ij}=1}} \text{clamp} \left(s_{\psi_2} - \psi_{ij}^d, \psi_{ij}^m - \psi_{ij}^d \right).
\end{aligned} \tag{63}$$

When the arguments of the clamp $(\cdot)$ function are vectors, its definition is equivalent to (37), such that

$$\text{clamp} \left(\mathbf{y}, \mathbf{a} \right) = \begin{cases} \mathbf{y} & \text{if} \quad \mathbf{y} \cdot \mathbf{a} \in \left(0, \|\mathbf{a}\|^2 \right] \\ \mathbf{0} & \text{otherwise.} \end{cases} \tag{64}$$

Note that, the geometrical considerations used with some of the terms from eq. (16) are reminiscent of those used for controlling purely bearing-based [21] and distance-based [28], [31], [32], [35] formations. We believe that our example can also be used as a template for applying the proposed technique to such systems.

IV. Simulation Testing

In order to test and showcase the presented technique, we have implemented a simulation of formation control according to the discussed assumptions, combined with a simulated relative localization and characteristics mimicking those measured in *UVDAR* [15]–[17]. In this simulation, virtual UAVs can set their velocity and observe their neighbors with noisy measurements of their relative positions and relative orientations about the vertical axis. No blind spots are considered to exist for the relative localization sensor. If one UAV is connected to another in a pre-defined observation graph, then it will successfully detect it in every measurement. Relative localization measurements are retrieved with a fixed rate. After each new measurement, every agent calculates the control action using eq. (63) and sets its velocity accordingly until the next measurement is obtained. Thus, between two consecutive measurements, the agents follow a linear trajectory and rotate at a constant rate. To explore how various formations are affected by our technique, we have employed four distinct formation scenarios, as seen in Fig. 12:

- two mutually observing UAVs,
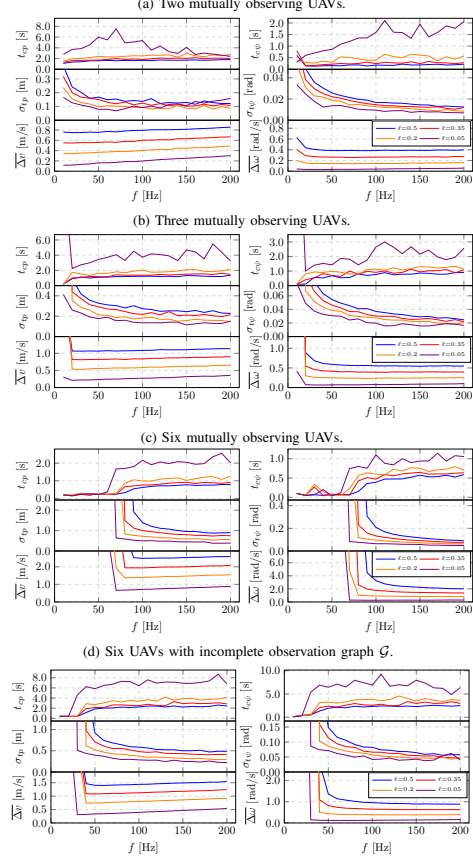
Fig. 11: Simulated formation flight of various configurations.
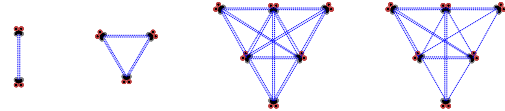
Fig. 12: Desired formations in our simulated experiments. The distance between the closest pair of agents is 5 m. The latter three formations are based on equilateral triangles. Blue lines denote observations between agents.

- three mutually observing UAVs with a fully connected observation graph $\mathcal{G}$ and the desired formation in the shape of an equilateral triangle,
- six mutually observing UAVs with a fully connected observation graph $\mathcal{G}$ and desired formation in the shape of a flat equilateral triangle, where three units represent its vertices and three units are in the center of each of its sides,
- six units with the same desired formation as above, with half of the edges in the observation graph $\mathcal{G}$ randomly removed such that the graph remains connected.

Each scenario was executed with four values of the parameter ℓ: 0.05, 0.2, 0.35, and 0.5. When $\ell = 0.5$, the control is equal to the pure proportional FEC, according to the eq. (16). The proportional factor k_e was set to 0.5. The simulated

measurement noise was set to emulate the properties of a real vision-based relative measurement system. Specifically, each measurement was burdened with noise sampled from a Gaussian probability distribution, such that:

- the distance component of the measured relative position had an additive noise with a standard deviation of 10% of the true distance;
- the bearing component had additive noise corresponding to $0.03\,\mathrm{rad}$;
- the relative heading measurement noise had a standard deviation of $0.26\,\mathrm{rad}$.

Additionally, each scenario was tested with measurement rates from $10\,\mathrm{Hz}$ to $200\,\mathrm{Hz}$ in increments of $10\,\mathrm{Hz}$. Because movement between the measurements is simulated with constant velocity, this change in rate can be alternatively interpreted as a change in the proportional factor k_{ef} from 0.05 to 0.0025, as shown in eq. (18).

In every case, the agents were initialized with random orientations and positions up to $20\,\mathrm{m}$ from the origin of the world coordinate frame. Every test was set to run for $\frac{2000}{f}\mathrm{s}$ in order to ensure that transient effects have passed. Each of these tests was evaluated according to three separate criteria for position (p) and orientation (ψ):

- Convergence time t_{cp} and $t_{c\psi}$ approximates the time it took to reach equilibrium. These values were obtained according to the definition in (45)
- The *stable-state* noise σ_{tp} and $\sigma_{t\psi}$ of the formation are obtained according to (46)
- Mean control-induced velocity and rotation rate change $\overline{\Delta v}$ and $\overline{\Delta \omega}$ are obtained according to (47)

The displacement from the 1D definition was substituted for the norm of the positional component of the formation error in case of the position. The results of these tests are shown in Fig. 11. Decreasing the maximal overshoot probability parameter ℓ generally decreases the mean velocity change $\overline{\Delta v}$ and mean rotation rate change $\overline{\Delta \omega}$, as well as the stable-state standard deviations σ_{tp} and $\sigma_{t\psi}$. This comes at the cost of an increase in the convergence time t_{cp} and $t_{c\psi}$.

However, if the measurements are obtained with a rate f that is too low, the agents do not achieve convergence at all, which is evident particularly in cases with six agents. They instead oscillate randomly, much like the behavior of a swarm. This effect is seen in the plots as very high stable-state errors and low convergence time at the low rate f. The convergence time is low for these cases, as the system converges to a chaotic state that does not improve over time. The agents in this state move in large linear steps, making them deviate too far from the region where the local linearization of the gradient according to eq. (12) is a reasonable approximation. The results also indicate that increasing the number of agents in the formation increases this effect, as seen in plots with more agents where the non-convergent behavior occurs with a higher f. If the number of agents remains unchanged, the same seems to be the case with more edges in the observation graph $\mathcal{G}$. A simple explanation is that with more agents, the error function $\mathbf{e}_F$ becomes increasingly non-linear w.r.t. to the local motion of an agent. Additionally, although more edges

Fig. 13: Three UAV platforms used in the real-world experiments.

in the observation graph lead to a more rigid formation in the theoretical noise-less case, with noise, more edges can lead to a more noisy formation control, which may be counter-intuitive. Because the agents do not share their measurements, each agent is also negatively influenced by noisy measurements of the other agents' poses. Similarly to a sum of Gaussian random variables, the resulting output has a larger uncertainty than each of the contributing measurements.

This also explains how the stable-state σ_{tp} and $\sigma_{t\psi}$ seem to increase with an increasing number of agents or observations, even if f is sufficiently high to avoid the aforementioned non-convergent behavior. Notably, Figures 11c and 11d show that convergence can be achieved at lower rates if the parameter ℓ is set to low values, illustrating an additional advantage of our technique for slower relative localization systems.

V. EXPERIMENTAL VERIFICATION

In order to verify that our algorithm is truly suitable for real-world robotic deployment, we have conducted a series of flights with three UAVs outdoors in an open space. Each unit was equipped with three UV-sensitive cameras, and UV LED markers comprising the full *UVDAR* system. These platforms are shown in Fig. 13. The implementation of the proposed control technique used for the experiments has been made available on-line[2]. It is based on the Robot Operating System (ROS) and the Multi-robot Systems Group (MRS) UAV system [36] that provides velocity-tracking functionality. The MRS UAV system relies on a SE(3) geometric controller for low-level control of the UAV and a Model Predictive Control (MPC) trajectory tracker that takes the desired trajectory (or in this case, desired velocity) and pre-shapes it to provide a dynamically feasible full-state reference for the SE(3) controller to execute.

In the experiments, we alternated between a series of the desired formations seen in Fig. 14, which we communicated to all three UAVs using a wireless network. The UAVs then applied the presented algorithm in combination with relative pose measurements from *UVDAR* to achieve that formation.

In order to showcase the need for addressing the effects of sensory noise in FEC, we have performed experiments with our modified action according to eq. (63), and with the original action without restraining according to eq. (16). We have also performed experiments where we alternated between the two modes in order to provide a more direct comparison between the two behaviors. For each flight, we have recorded the agent positions derived from the precise GNSS-RTK system used as a ground truth only for evaluation purposes. The following values are included in the plots:

[2]`https://github.com/ctu-mrs/difec-ron`

13
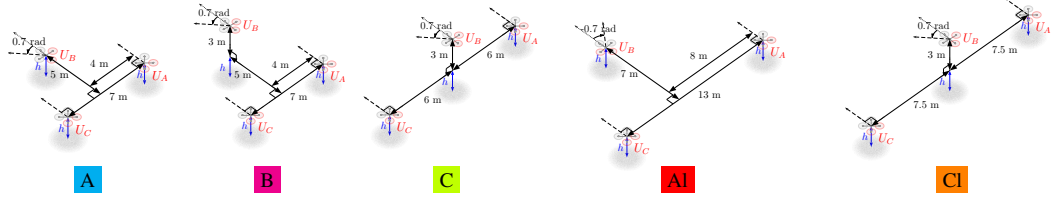
Fig. 14: The desired formations used in our experimental verification. The red propellers denote the tail side of the UAVs. The larger formations Al and Cl contain mutual distances at which *UVDAR*, at the used setting, can not reliably provide direct localization between UAV U_A and UAV U_C, such that the formation is "held together" through UAV U_B.
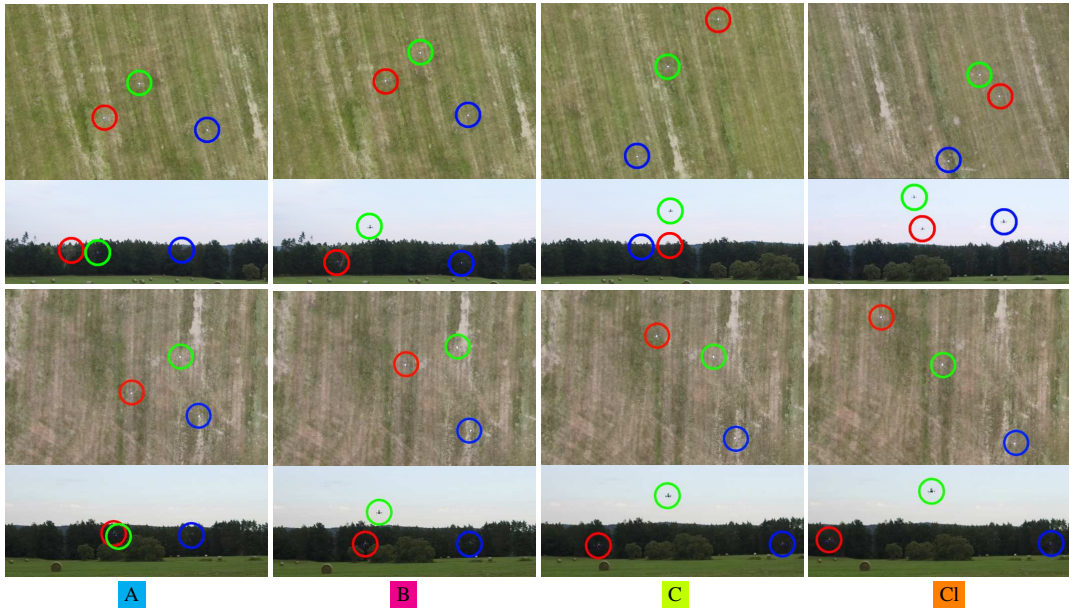
Fig. 15: Flight with four desired formations. Photographs of the top and side view of the formation are from the time of the closest convergence of each case. The top line of pictures presents flights without the proposed restraining technique, and flights with restraining are at the bottom. Without restraining, the formation significantly drifted and rotated in the third desired formation due to an accumulation of noise. It also took significantly longer for convergence to occur for this formation. The fourth desired formation was only achievable with our restraining technique applied. In the case without restraining, chaotic movement occurred and the achieved formation differed significantly from the desired one.

- The norm e_F of the error vector $\mathbf{e}_F$ from eq. (11), which was obtained using absolute positions.
- The UAV-wise average norm e_p of the relative position difference between individual current and desired relative positions of neighbors.
- The UAV-wise average difference e_ψ between individual current and desired relative orientations of neighbors.
- The UAV-wise average angular velocity v_ψ where higher values are detrimental for visual relative localization.
- The UAV-wise average positional acceleration a_p where higher values are detrimental for relative pose estimation.
- The connectivity of the observation graph $\mathcal{G}$, expressed through the Fiedler eigenvalue f of the Laplace matrix L of this graph.

A. Small formations

In these experiments, we have alternated between formations A, B, and C, as shown in Fig. 14. Formation C has a relatively large distance between UAVs A and C, leading to greater noise in estimation of both relative position and relative orientation. First, we have executed the sequence with restraining activated according to eq. (63). The results of this test are shown in Fig. 16a. The proportional factor k_e was set to 0.06, the sensor sampling frequency to 10 Hz, and the threshold ℓ was set to 0.3 - a value we empirically determined to be an adequate compromise between convergence speed and restraining reaction to sensory noise. Each time a new desired formation was selected, the actual formation converged smoothly towards it.

We have then performed this scenario with restraining disabled, while only activating at specific intervals to show that doing so stabilized the oscillating motion of the UAVs. The results of this experiment are shown in Fig. 16b, with darker areas signifying where restraining was activated. The formation had a significantly worse rate of convergence to the desired formation, as well as exhibiting more unnecessary accelerations and rotations. In the sections where restraining was activated, this effect is visibly mitigated.

14
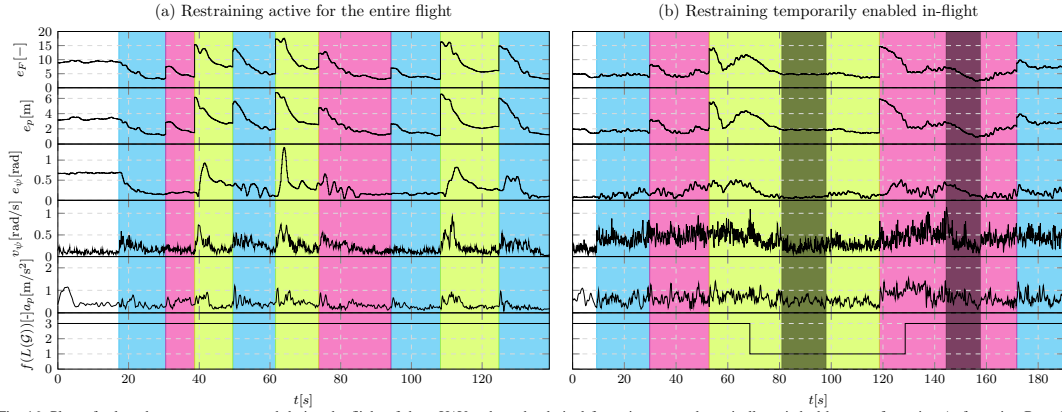
Fig. 16: Plots of selected parameters measured during the flight of three UAVs where the desired formations were dynamically switched between formation A, formation B, and formation C. In the left plot, restraining was active during the entire flight time. In the right plot, restraining was disabled for most of the flight and enabled for the intervals marked by the darker color. Note, how enabling restraining rapidly improves the behavior parameters of the formations.
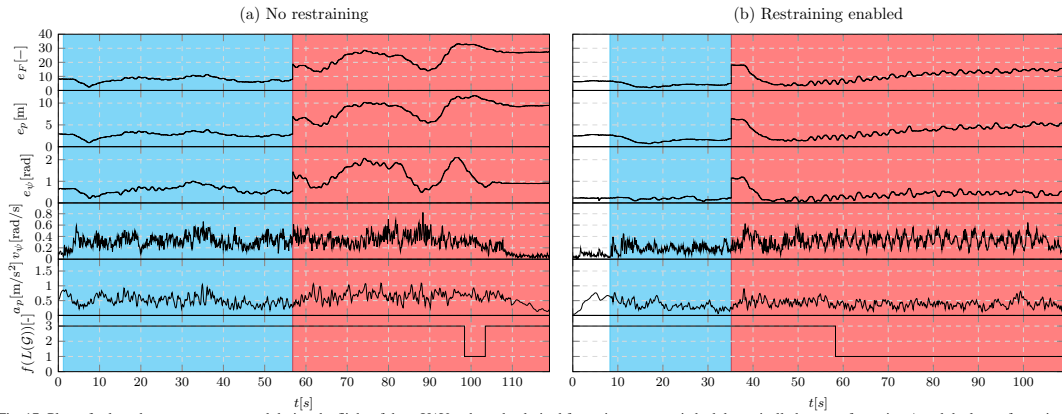
Fig. 17: Plots of selected parameters measured during the flight of three UAVs where the desired formations were switched dynamically between formation A and the larger formation formation Al. Note that, without the proposed restraining technique, the formation rapidly loses adherence to the desired shape.
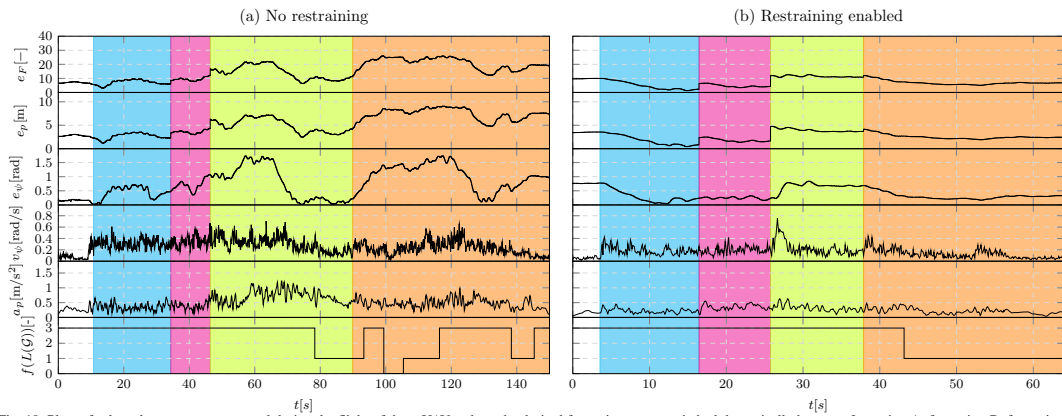
Fig. 18: Plots of selected parameters measured during the flight of three UAVs where the desired formations were switched dynamically between formation A, formation B, formation C, and the larger formation Cl. Without the proposed restraining technique, the UAVs occasionally lost contact while following the largest desired formation Cl. This loss of contact occurred due to the UAVs reaching undesirably large distances from each other as a consequence of measurement noise and their individual disordered motion. The proposed restraining technique enabled formation preservation, even in these challenging sensory conditions.

15

B. Large formations

To test the real-world limits of the proposed algorithm, we have performed two sequences that end in the more extreme formations Al and Cl. Formation Al contains distances between the UAVs at which *UVDAR* performs poorly, and its ability to estimate the relative orientation of the target shows significant noise. Formation Cl contains the distance between UAVs U_A and U_C at the limit of the *UVDAR* operational range, leading to a state where the two are only held in formation by mutual visibility with UAV U_B.

We have performed two scenarios, each with FEC according to eq. (16) and with restraining according to eq. (63), where the proportional factor k_e was set to 0.06, the average sensor sampling frequency at 10 Hz, and the threshold ℓ set to 0.3.

In the first scenario, we set formation A as the desired formation. Once it converged, we set the desired formation to Al. The results of this experimental scenario are shown in Fig. 17. It is evident from the plots that without restraining, the system took significantly longer for convergence to the desired formation A, and completely failed to converge to the desired formation Al due to the larger measurement noise at greater relative distances.

In the experiment with restraining, the periodic oscillations seen superposed on the region for formation Al are caused by the central UAV U_B oscillating between minimizing the formation error with respect to UAV U_A and UAV U_C. This is due to UAV U_B connecting the other two UAVs by mutual observations while they individually drift in different directions. This drift was the result of both neighbors observing UAV U_B with high measurement noise, especially in the relative heading due to the distance. Despite this detrimental behavior and under all considered aspects, the overall characteristics of this flight are better than without restraining.

The second scenario comprised of a sequence of desired formations A, B, C, and Cl, with and without restraining. The plots of the results from these experiments are shown in Fig. 18. It is clearly visible in these plots that not only were the accelerations and angular velocities significantly reduced with restraining, but convergence to the desired formations was also more precise in comparison with the experiment using the original approach. With restraining applied, convergence to the desired formation was achieved, even in cases where the convergence entirely failed otherwise. Further, a shorter time scale was required for convergence in the case with restraining enabled. The unrestrained FEC even caused the observation graph to disconnect at one point, evidenced by the Fiedler eigenvalue dropping to zero.

These experiments are shown in the video documentation[3]. The most significant results of the three experimental scenarios are summarized in Table I, where we show the average rotation rates and acceleration for all three UAVs during each scenario. The error change rate was averaged from the whole active FEC period, excluding the instances of desired formation switching. For the *Small formations* flight without restraining, we have excluded the data where restraining was temporarily activated, seen as dark areas in figure 16. These results clearly show that

[3] `http://mrs.felk.cvut.cz/difec-ron`

Flight	Value		Without restraining	With restraining	Improv.
Small formations	$\overline{v_\psi}$	$\left[\text{rad/s}\right]$	0.4096	0.2056	49.8%
	$\overline{a_p}$	$\left[\text{m/s}^2\right]$	0.6403	0.3558	44.4%
A B C	$\overline{de_p/dt}$	$\left[\text{m/s}\right]$	−0.0420	−0.2085	0.167m/s
Large formations 1	$\overline{v_\psi}$	$\left[\text{rad/s}\right]$	0.2769	0.2686	3.0%
	$\overline{a_p}$	$\left[\text{m/s}^2\right]$	0.5142	0.3759	26.9%
A Al	$\overline{de_p/dt}$	$\left[\text{m/s}\right]$	0.0146	−0.0190	0.034m/s
Large formations 2	$\overline{v_\psi}$	$\left[\text{rad/s}\right]$	0.2579	0.1571	39.1%
	$\overline{a_p}$	$\left[\text{m/s}^2\right]$	0.4965	0.2436	50.9%
A B C Cl	$\overline{de_p/dt}$	$\left[\text{m/s}\right]$	0.0144	−0.0737	0.088m/s

TABLE I: Improvement evaluation for the three experimental flight scenarios. Average angular velocities and positional accelerations, as well as the rate of change in average positional formation error, are shown. For all three variables, lower values are better. The average acceleration values are a metric that shows the amount of oscillations in the system, where a higher value leads to degradation of the onboard sensing performance. These values show how the presented technique significantly reduces the tilting and accelerations negatively affecting computer vision performance. It is also evident that the UAV team had a significantly increased ability to converge to the desired formation, as seen by the lower error change rate.

the proposed technique improves FEC both in terms of reduced tilting, which is beneficial for computer vision performance, as well as the overall ability of a UAV team to converge to a desired formation, especially in cases of significantly noisy relative pose measurements.

In the video documentation, the overall formation exhibits drifting over time, as is expected since the FEC does not account for the positions of agents in the world. Modification of the original cost function that generated the FEC to contain terms penalizing agent motion is planned for future work.

VI. CONCLUSION

In this work, we present a novel technique to allow real-world deployment of the classical distributed formation-enforcing control (DIFEC) based on relative neighbor observation derived from the graph rigidity theory. This technique, called *restraining*, addresses the problems inherent in such formation control if the relative measurements by onboard sensors are obtained in discrete time-steps, and are subject to noise with a known distribution. Since most of the available onboard relative localization sensors provide discrete data with known characteristics, this technique has wide range of applicability. Our proposal involves designing of the control action, such that we can exploit the knowledge of the statistical properties of the observation noise.

The user of a multi-robot system utilizing the proposed technique can specify a level of probability ℓ for the UAVs to exceed the desired formation pose during the performed linear motions based on the current relative measurement.

We have applied our technique to formations based on the relative pose observations comprising the 3D positions and headings of UAVs. However, this general principle can also be applied to other situations, such as bearing-based or distance-based formations, and even to tasks beyond simple formation enforcement.

We have verified the performance of the presented technique both in simulation and in real-world experiments with the use of the *UVDAR* sensor. All various experiments demonstrated significant improvement in flight characteristics - see Table I - and also numerous situations where formation flight fails without the proposed technique.

CHAPTER 4. RESEARCH MOTIVATED BY THE *UVDAR* SYSTEM

APPENDIX

A. Formation enforcing action in $\mathbb{R}^3 \times \mathbf{S}^1$

We will derive a control action that moves the current formation down the gradient of the overall error of the formation. As established in section II, such control can be expressed as

$$\dot{\mathbf{q}} = \begin{bmatrix} \mathbf{p} \end{bmatrix} k_e \left(\frac{\partial \varkappa_{\mathcal{G}}(\mathbf{q})}{\partial \mathbf{q}} \right)^T \mathbf{e}_F. \tag{65}$$

We must turn the above into control equations usable in a group of UAVs that do not explicitly communicate their poses and individually rely only on their observations. Following similar steps to those used to obtain bearing-based FEC in [22], we define the *relative pose rigidity matrix* $\mathbf{H}_{\mathcal{G}}^W$:

$$\mathbf{H}_{\mathcal{G}}^W(\mathbf{q}) \triangleq \frac{\partial \varkappa_{\mathcal{G}}(\mathbf{q})}{\partial \mathbf{q}}. \tag{66}$$

This matrix expresses how much the formation $\varkappa_{\mathcal{G}}(\mathbf{q})$ changes with the agent poses $\mathbf{q}$. Thereafter, the four lines of $\mathbf{H}_{\mathcal{G}}^W$ corresponding to the edge $e = (i, j)$ are

$$\mathbf{H}_{\mathcal{G}ij}^W(\mathbf{q}) = \begin{bmatrix} \text{-0-} & -\mathbf{R}(\psi_i)^T & \frac{\partial \mathbf{R}(\psi_i)^T}{\partial \psi_i}\mathbf{p}_{ij}^W & \text{-0-} & \mathbf{R}(\psi_i)^T & \mathbf{0} & \text{-0-} \\ \text{-0-} & \underbrace{0}_{\partial \varkappa_{\mathcal{G}}/\partial \mathbf{p}_i} & \underbrace{-1}_{\partial \varkappa_{\mathcal{G}}/\partial \psi_i} & \text{-0-} & \underbrace{0}_{\partial \varkappa_{\mathcal{G}}/\partial \mathbf{p}_j} & \underbrace{1}_{\partial \varkappa_{\mathcal{G}}/\partial \psi_j} & \text{-0-} \end{bmatrix}. \tag{67}$$

This expression can be transformed into the local frame of the agent i, which is then observing agents j through relative pose measurements. This transformation yields the matrix $\mathbf{H}_{\mathcal{G}}^l$ and comprises of per-edge sub-matrices as follows:

$$\mathbf{H}_{\mathcal{G}ij}^l(\varkappa_{\mathcal{G}}) = \begin{bmatrix} \text{-0-} & -\mathbf{I}_3 & \mathbf{S}^T\mathbf{p}_{ij} & \text{-0-} & -\mathbf{R}(\psi_{ij}) & \mathbf{0} & \text{-0-} \\ \text{-0-} & 0 & -1 & \text{-0-} & 0 & 1 & \text{-0-} \end{bmatrix} \tag{68}$$

where $\mathbf{S} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$, $\mathbf{p}_{ij}$ is the relative position of agent j measured by agent i in its own body frame, and ψ_{ij} is the relative heading of agent j measured by i. Note that, the columns corresponding to changes in $\mathbf{p}_j$ in the world frame were also rotated to express their motion w.r.t. the perspective of agent i. The action $\dot{\mathbf{q}}$ viewed in the local frames of each individual agent is converted to $\dot{\mathbf{q}}^l$, resolving to

$$\dot{\mathbf{q}}^l = k_e \mathbf{H}_{\mathcal{G}}^{l\,T}\left(\varkappa_{\mathcal{G}}(\mathbf{q}_d) - \varkappa_{\mathcal{G}}(\mathbf{q})\right) \\ = [[\mathbf{u}_1{}^T, \omega_1], [\mathbf{u}_2{}^T, \omega_2], ..., [\mathbf{u}_n{}^T, \omega_n]]^T. \tag{69}$$

If we presume the relative observation graph $\mathcal{G}$ to be connected, we can construct a decentralized control scheme that will enforce the desired formation shared by all observing agents. The resulting scheme corresponds to eq. (13):

$$\mathbf{u}_i = k_e \left(\sum_{j \in \mathcal{N}_i} c_{ij}\left(\mathbf{p}_{ij} - \mathbf{p}_{ij}^d\right) - \sum_{j \in \mathcal{N}_i} c_{ji}\mathbf{R}(\psi_{ji})^T\left(\mathbf{p}_{ji} - \mathbf{p}ji^d\right) \right),$$

$$\omega_i = k_e \left(-\sum_{j \in \mathcal{N}_i} c_{ij}\left(\mathbf{p}_{ij}^T\mathbf{S}\left(\mathbf{p}_{ij} - \mathbf{p}_{ij}^d\right)\right) \right. \\ \left. + \sum_{j \in \mathcal{N}_i} c_{ij}\left(\psi_{ij} - \psi_{ij}^d\right) - \sum_{j \in \mathcal{N}_i} c_{ji}\left(\psi_{ji} - \psi_{ji}^d\right) \right).$$

B. Stability analysis

In order to analyze the stability of the FEC system obtained in section II, we propose the following Lyapunov function:

$$V(\mathbf{e}_F) = \mathbf{e}_F{}^T \mathbf{P}\mathbf{e}_F \tag{70}$$

where $\mathbf{P}$ is a positive-definite matrix. The time-derivative of V is

$$\dot{V} = \dot{\mathbf{e}}_F{}^T\mathbf{P}\mathbf{e}_F + \mathbf{e}_F{}^T\mathbf{P}\dot{\mathbf{e}}_F. \tag{71}$$

The error dynamics $\dot{\mathbf{e}}_F$ are obtained as

$$\dot{\mathbf{e}}_F = \frac{\partial \mathbf{e}_F}{\partial t} = \frac{\partial \mathbf{e}_F}{\partial \varkappa_{\mathcal{G}}} \frac{\partial \varkappa_{\mathcal{G}}}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial t}. \tag{72}$$

Using (12) and (66), the above resolves to

$$\dot{\mathbf{e}}_F = -\mathbf{I_4}\mathbf{H}_{\mathcal{G}}^W k_e \mathbf{H}_{\mathcal{G}}^{W\,T}\mathbf{e}_F. \tag{73}$$

For notational simplicity, we will define $\mathbf{M} \triangleq \mathbf{H}_{\mathcal{G}}^W\mathbf{H}_{\mathcal{G}}^{W\,T}$ and simplify the above to

$$\dot{\mathbf{e}}_F = -k_e\mathbf{M}\mathbf{e}_F. \tag{74}$$

Plugging the above into (71) yields

$$\dot{V} = -k_e\left(\mathbf{e}_F{}^T\mathbf{M}\mathbf{P}\mathbf{e}_F + \mathbf{e}_F{}^T\mathbf{P}\mathbf{M}\mathbf{e}_F\right) \\ = -k_e\mathbf{e}_F{}^T\left(\mathbf{M}\mathbf{P} + \mathbf{P}\mathbf{M}\right)\mathbf{e}^F. \tag{75}$$

For the system to be stable in some neighborhood $\mathcal{B}$ of equilibrium $\mathbf{e}_F = \mathbf{0}$, it must hold that

$$\dot{V} < 0 \quad \forall \mathbf{e}_F \in \mathcal{B}\backslash\{\mathbf{0}\}, \tag{76}$$

therefore $(\mathbf{M}\mathbf{P} + \mathbf{P}\mathbf{M})$ must be a positive definite matrix. This condition is difficult to evaluate for a general case, since $\mathbf{M}$ depends on the observation graph, desired formation, and current state $\mathbf{q}$. We will evaluate the condition for a basic example formation of two agents where agent 1 observes agent 2. In this case,

$$\mathbf{H}_{\mathcal{G}}^W = \begin{bmatrix} -\mathbf{R}(\psi_1)^T & \mathbf{S}^T\mathbf{p}_{12}^W & \mathbf{R}(\psi_1)^T & \mathbf{0_3} \\ \mathbf{0_3}^T & -1 & \mathbf{0_3}^T & 1 \end{bmatrix}, \tag{77}$$

and therefore $\mathbf{M}$ is

$$\mathbf{M} = \begin{bmatrix} 2+\left(\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_2\right)^2 & -\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_2\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_1 & 0 & \begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_2 \\ -\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_2\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_1 & 2+\left(\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_1\right)^2 & 0 & -\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_1 \\ 0 & 0 & 2 & 0 \\ \begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_2 & -\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_1 & 0 & 1 \end{bmatrix}. \tag{78}$$

Since $\mathbf{M}$ is symmetric, if its leading principal minors are positive, then it is positive definite. To prove this is the case, we evaluate all four leading principal minors. These are

$$m_1 = 2 + \left(\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_2\right)^2$$
$$m_2 = 4 + 2\left(\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_2\right)^2 + 2\left(\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_1\right)^2$$
$$m_3 = 8 + 4\left(\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_2\right)^2 + 4\left(\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_1\right)^2 \tag{79}$$
$$m_4 = 16 + 4\left(\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_2\right)^2 + 4\left(\begin{bmatrix}\mathbf{p}_{12}^W\end{bmatrix}_1\right)^2$$

In the above, m_o denotes the leading principal minor corresponding to the determinant of the upper-left submatrix of size $o \times o$. All of the above leading principal minors are positive

for $\mathbf{p}_{12}^W \in \mathbb{R}^3$, and thus M is a positive definite matrix. Therefore, choosing trivial $\mathbf{P} = \mathbf{I_4}$ for (75), condition (76) holds for arbitrary $\mathcal{B}$ and thus the system is stable, converging to $\mathbf{e}_F = \mathbf{0}$.

The above can not be trivially generalized to an arbitrary formation. In general, $\mathbf{M}$ will have the format

$$\mathbf{M} = \mathbf{H}\mathbf{H}^T = \begin{bmatrix} \mathbf{E}_1 \\ \mathbf{E}_2 \\ \vdots \end{bmatrix} \begin{bmatrix} \mathbf{E}_1{}^T & \mathbf{E}_2{}^T & \dots \end{bmatrix} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} & \dots \\ \mathbf{E}_{21} & \mathbf{E}_{22} & \dots \\ \vdots & \vdots & \ddots \end{bmatrix}, \quad (80)$$

where $\mathbf{E}_a$ is a set of four rows in H corresponding to the observation graph edge a in the form of (67), and we define $\mathbf{E}_{ab} \triangleq \mathbf{E}_a \mathbf{E}_b{}^T$ which is a symmetrical 4×4 matrix. When the edges a and b do not share a vertex (an agent), then $\mathbf{E}_{ab} = \mathbf{0_4}$. Each $\mathbf{E}_{aa}$ has the format of (78), substituting vertex indices 1 and 2 for the indices of vertices connected by edge a. If edges a and b both lead *out of* their shared vertex, $\mathbf{E}_{ab}$ has the format

$$\mathbf{E}_{ab} = \begin{bmatrix} 1 + \left([\mathbf{p}_a^W]_2 \right)^2 & -[\mathbf{p}_a^W]_2 [\mathbf{p}_a^W]_1 & 0 & [\mathbf{p}_a^W]_2 \\ -[\mathbf{p}_a^W]_2 [\mathbf{p}_a^W]_1 & 1 + \left([\mathbf{p}_a^W]_1 \right)^2 & 0 & -[\mathbf{p}_a^W]_1 \\ 0 & 0 & 1 & 0 \\ [\mathbf{p}_a^W]_2 & -[\mathbf{p}_a^W]_1 & 0 & 1 \end{bmatrix}, \quad (81)$$

where $\mathbf{p}_a$ is the relative position represented by the observation edge a. If edges a and b both lead *into* their shared vertex, $\mathbf{E}_{ab}$ has the format of 4×4 identity matrix

$$\mathbf{E}_{ab} = \mathbf{I_4}. \quad (82)$$

If edge a leads from vertex i *into* vertex j and edge b leads *out of* vertex j, $\mathbf{E}_{ab}$ has the format

$$\mathbf{E}_{ab} = \begin{bmatrix} -R\left(\psi_j\right)^T R(\psi_i) & \begin{bmatrix} -[\mathbf{p}_a^W]_2 \\ [\mathbf{p}_a^W]_1 \\ 0 \end{bmatrix} \\ [0 \; 0 \; 0] & -2 \end{bmatrix}. \quad (83)$$

Lastly, if edges a and b connect vertices i and j in mutually opposite directions, $\mathbf{E}_{ab}$ has the format

$$\mathbf{E}_{ab} = \begin{bmatrix} -R(\psi_i)^T R(\psi_j) - R(\psi_j)^T R(\psi_i) & \begin{bmatrix} -[\mathbf{p}_a^W]_2 \\ [\mathbf{p}_a^W]_1 \\ 0 \end{bmatrix} \\ \left[-[\mathbf{p}_b^W]_2 \; [\mathbf{p}_b^W]_1 \; 0 \right] & -2 \end{bmatrix}. \quad (84)$$

The above can serve as building blocks for evaluating the stability of various relative pose-based formations, and general analysis is out of the scope of this work.

C. 1 active agent and 1 passive agent

We will analyze the performance of our controller in a simplified scenario with two unidimensional agents with positions $p_1[k], p_2[k] \in \mathbb{R}$. Agent 2 is static while Agent 1 implements our robust controller, so that its relative position to Agent 2 ($p_{12}[k] \triangleq p_2 - p_1[k]$) reaches a desired relative position p_{12}^d. We also introduce the displacement $\Delta_{12}[k] \triangleq p_{12}[k] - p_{12}^d$. Agent 1 has a noisy unbiased measurement $p_{12}^m[k] \triangleq p_{12}[k] + e_{12}[k]$ with $e_{12}[k] \sim \mathcal{N}(0, \sigma_{12}[k]^2)$, and also has an estimate of $\sigma_{12}[k]$ denoted $\hat{\sigma}_{12}[k]$. We will assume that $\hat{\sigma}_{12}[k] = \sigma_{12}[k]$. Due to the error, we have a measure of the

displacement $\delta_{12}[k] \triangleq \Delta_{12}[k] + e_{12}[k]$. The position of Agent 1 is then given by:

$$p_1[k+1] = k_{ef} d_1[k] + p_1[k],$$
$$d_1[k] = \text{clamp}(y_{12}[k], p_{12}[k]^m - p_{12}^d)$$
$$= \begin{cases} y_{12}[k] & \text{if } |\delta_{12}[k]| > -\hat{\sigma}_{12}[k]\Phi^{-1}(\ell), \\ 0 & \text{otherwise} \end{cases} \quad (85)$$
$$y_{12}[k] = \text{sign}\left(\delta_{12}[k]\right)\hat{\sigma}_{12}[k]\Phi^{-1}(\ell) + \delta_{12}[k]$$

We assume $\ell \in (0, 0.5]$, and thus $\Phi^{-1}(\ell) \leq 0$. Let us start by calculating the conditional probability that the agent does not move at time instant k, given $\Delta_{12}[k]$. We refer to this probability as the conditional stopping probability:

$$q_1[k](\Delta_{12}[k]) \triangleq P\left(d_1[k] = 0 \,|\, \Delta_{12}[k]\right)$$
$$= P\left(|\Delta_{12}[k] + e_{12}[k]| \leq -\hat{\sigma}_{12}[k]\Phi^{-1}(\ell)\right) \quad (86)$$

Since $e_{12}[k]$ follows a Gaussian distribution, and we are assuming $\hat{\sigma}_{12}[k] = \sigma_{12}[k]$, then it follows that:

$$q_1[k](\Delta_{12}[k]) = \Phi\left(-\Delta_{12}[k]/\hat{\sigma}_{12}[k] - \Phi^{-1}(\ell)\right)$$
$$- \Phi\left(-\Delta_{12}[k]/\hat{\sigma}_{12}[k] + \Phi^{-1}(\ell)\right). \quad (87)$$

Note that, the arguments of the right-hand side terms of (87) are antisymmetric. This stopping probability is caused by the clamping function. From Fig. 19, we can see that clamping introduces a spatial probabilistic motion filter. The motion probability of the agent decreases as $|\Delta_{12}[k]|$ becomes smaller, with the minimum motion probability:

$$\bar{q}_1[k](0) = 2\ell, \quad (88)$$

where $\bar{q}_1[k](\Delta_{12}[k]) = 1 - q_1[k](\Delta_{12}[k])$. We observe that the effect of this spatial probabilistic motion filter becomes negligible for $|\Delta_{12}[k]'| \geq -\Phi^{-1}(\ell) + 3$ where $\Delta_{12}[k]' \triangleq \frac{\Delta_{12}[k]}{\hat{\sigma}_{12}[k]}$. Thus, we can approximate the nonlinear system (85) with the switched system (89).

$$p_1[k+1] = \begin{cases} k_{ef} \text{clamp}(y_{12}[k], p_{12}[k]^m - p_{12}^d) + p_1[k] \\ \qquad \text{if } |\Delta_{12}[k]'| \leq -\Phi^{-1}(\ell) + 3, \quad (89) \\ k_{ef} y_{12}[k] + p_1[k] \quad \text{otherwise}, \end{cases}$$

When the agent is close to reaching the target p_{12}^d (i.e., when $\Delta_{12}[k]' \in \mathcal{S}_\ell$ where $S_\ell = \left[\Phi^{-1}(\ell) - 3, -\Phi^{-1}(\ell) + 3\right]$), then the switched system (89) behaves as the nonlinear system (85). Otherwise, the switched system (89) behaves like a linear system.

1) Linear System: When the switched system operates as a linear system, we have:

$$p_1[k+1] =$$
$$= k_{ef}\left(\text{sign}\left(\delta_{12}[k]\right)\hat{\sigma}_{12}[k]\Phi^{-1}(\ell) + \delta_{12}[k]\right) + p_1[k],$$
$$= k_{ef}\left(\text{sign}\left(\Delta_{12}[k] + e_{12}[k]\right)\hat{\sigma}_{12}[k]\Phi^{-1}(\ell) + \Delta_{12}[k]\right)$$
$$+ k_{ef}e_{12}[k] + p_1[k],$$

since $p_2 - p_1[k] - p_{12}^d = \Delta_{12}[k]$, then:

$$\Delta_{12}[k+1] = -k_{ef} \,\text{sign}\left(\Delta_{12}[k] + e_{12}[k]\right)\hat{\sigma}_{12}[k]\Phi^{-1}(\ell)$$
$$- k_{ef}e_{12}[k] + (1 - k_{ef})\Delta_{12}[k],$$

18

We divide the equation by $\hat{\sigma}_{12}[k]$:

$$\frac{\Delta_{12}[k+1]}{\hat{\sigma}_{12}[k]} = -k_{ef}\,\mathrm{sign}\,(\Delta_{12}[k] + e_{12}[k])\Phi^{-1}(\ell)$$
$$- \frac{k_{ef}e_{12}[k]}{\hat{\sigma}_{12}[k]} + (1 - k_{ef})\frac{\Delta_{12}[k]}{\hat{\sigma}_{12}[k]},$$

Now, $\sigma_{12}[k]$ varies slowly with the distance $|\Delta_{12}[k]|$, and thus we can say that the standard deviation of the error for two consecutive instants is approximately constant, i.e., $\sigma_{12}[k] \approx \sigma_{12}[k + 1]$. Therefore, $\frac{\Delta_{12}[k+1]}{\hat{\sigma}_{12}[k]} \approx \frac{\Delta_{12}[k+1]}{\hat{\sigma}_{12}[k+1]} = \Delta'_{12}[k+1]$. Lastly, we have:

$$\Delta'_{12}[k+1] = -k_{ef}\,\mathrm{sign}\,(\Delta_{12}[k] + e_{12}[k])\Phi^{-1}(\ell)$$
$$- \frac{k_{ef}e_{12}[k]}{\hat{\sigma}_{12}[k]} + (1 - k_{ef})\Delta'_{12}[k],$$

We then take the expected value w.r.t. the measurement noise

$$\mathbb{E}[\Delta'_{12}[k+1]] = -k_{ef}\mathbb{E}[\mathrm{sign}\,(\Delta_{12}[k] + e_{12}[k])]\Phi^{-1}(\ell)$$
$$+ (1 - k_{ef})\mathbb{E}[\Delta_{12}[k]'],$$

where

$$\mathbb{E}[\mathrm{sign}\,(\Delta_{12}[k] + e_{12}[k])] = \mathrm{sign}(\Delta_{12}[k])\bar{p}_e(k) \\ - \mathrm{sign}(\Delta_{12}[k])p_e(k), \quad (90)$$

and

$$p_e(k) = \mathrm{P}\,(\mathrm{sign}(\Delta_{12}[k]) \neq \mathrm{sign}(\Delta_{12}[k] + e_{12}[k])) \\ = \mathrm{P}\left(\frac{\Delta_{12}[k] + e_{12}[k]}{\Delta_{12}[k]} < 0\right) = 1 - \Phi\left(\left|\Delta'_{12}[k]\right|\right) \quad (91)$$

However, this system is valid only for $\Delta'_{12}[k] \notin \mathcal{S}_\ell$ and under this condition, $p_e(\Delta_{12}[k], k) \ll 1$. As a consequence, $\mathbb{E}[\mathrm{sign}\,(\Delta_{12}[k] + e_{12}[k])] \approx \mathrm{sign}(\Delta_{12}[k])$. In addition, since $\hat{\sigma}_{12}[k] > 0$, then $\mathrm{sign}(\Delta_{12}[k]) = \mathrm{sign}(\Delta'_{12}[k])$. Therefore,

$$\mathbb{E}\left[\Delta'_{12}[k+1]\right] \approx -k_{ef}\,\mathrm{sign}\,(\Delta'_{12}[k])\Phi^{-1}(\ell) \\ + (1 - k_{ef})\mathbb{E}\left[\Delta'_{12}[k]\right], \quad (92)$$

Next, we write the iterative eq. (92) as a function of the initial state $\Delta'_{12}[0]$:

$$\mathbb{E}\left[\Delta'_{12}[k+1]\right] = -k_{ef}\Phi^{-1}(\ell)\sum_{n=1}^{k}\mathrm{sign}(\Delta'_{12}[n])(1-k_{ef})^{n-1} \\ + (1 - k_{ef})^{k+1}\Delta'_{12}[0]. \quad (93)$$

Since this system is valid for $\Delta'_{12}[k] \notin \mathcal{S}_\ell$ and $0 \in \mathcal{S}_\ell$, we then have that $\mathrm{sign}(\Delta'_{12}[k]) = \mathrm{sign}(\Delta'_{12}[0])$ for all k, as long as this system is valid. Thus, we obtain the following linear system:

$$\mathbb{E}\left[\Delta'_{12}[k+1]\right] = -k_{ef}\,\mathrm{sign}(\Delta'_{12}[0])\Phi^{-1}(\ell)\sum_{n=1}^{k}(1-k_{ef})^{n-1} \\ + (1 - k_{ef})^{k+1}\Delta'_{12[0]}, \quad (94)$$

and using the geometric series formula, we obtain:

$$\mathbb{E}\left[\Delta'_{12}[k+1]\right] = -\mathrm{sign}(\Delta'_{12[0]})\Phi^{-1}(\ell)(1 - (1 - k_{ef})^{k+1}) \\ + (1 - k_{ef})^{k+1}\Delta'_{12[0]}. \quad (95)$$

If $k_{ef} \in (0, 2)$, then $|1 - k_{ef}| < 1$ and the system is stable. Consequently, we have:

$$\lim_{k \to +\infty} \mathbb{E}\left[\Delta'_{12}[k+1]\right] = -\mathrm{sign}(\Delta'_{12}[0])\Phi^{-1}(\ell) \quad (96)$$

and $-\mathrm{sign}(\Delta'_{12}[0])\Phi^{-1}(\ell) \in \mathcal{S}_\ell$. This implies that if $\Delta'_{12}[0] \notin \mathcal{S}_\ell$ and the system is stable, then for some value k, we will have $\Delta'_{12}[k] \in \mathcal{S}_\ell$. This demonstrates that if the system is stable. If it is operating in the linear region, then it will enter into the clamped region eventually.

2) Nonlinear System: Now, let us focus on $\Delta_{12}[k] \in \mathcal{S}_\ell$. During this phase, the standard deviation $\sigma_{12}[k]$ of the measurement error remains approximately constant, since the motion of Agent 1 is small w.r.t. its distance to Agent 2. We will denote this final value of $\sigma_{12}[k]$ as σ.

$$\Delta_{12}[k+1] = \Delta_{12}[k] \\ - k_{ef}\mathrm{clamp}(\mathrm{sign}(\delta_{12}[k])\sigma\Phi^{-1}(\ell) + \delta_{12}[k], p_{12}[k]^m - p_{12}^d) \quad (97)$$

Then, we take the expected value w.r.t. the measurement noise:

$$\mathbb{E}\left[\Delta_{12}[k+1]\right] = \mathbb{E}\left[\Delta_{12}[k]\right] \\ - k_{ef}\mathbb{E}\left[\mathrm{clamp}(\mathrm{sign}(\delta_{12}[k])\sigma\Phi^{-1}(\ell) + \delta_{12}[k], p_{12}[k]^m - p_{12}^d)\right] \quad (98)$$

which becomes:

$$\mathbb{E}\left[\Delta_{12}[k+1]\right] = \mathbb{E}\left[\Delta_{12}[k]\right] \\ + k_{ef}\mathbb{E}_k\left[\int_{-\infty}^{L_-}\left(\sigma\Phi^{-1}(\ell) - \Delta_{12}[k] - x\right)f_e(x)\mathrm{d}x\right] \\ - k_{ef}\mathbb{E}_k\left[\int_{L_+}^{\infty}\left(\sigma\Phi^{-1}(\ell) + \Delta_{12}[k] + x\right)f_e(x)\mathrm{d}x\right] \quad (99)$$

where $L_- = -\Delta_{12}[k] + \sigma\Phi^{-1}(\ell)$ and $L_+- = -\Delta_{12}[k] - \sigma\Phi^{-1}(\ell)$. $\mathbb{E}[\Delta_{12}[k]]$ is the expected value w.r.t. all the measurement errors (i.e., $e_{12[0]}, e_{12[1]}, \cdots, e_{12}[k]$), but $\mathbb{E}_k[\Delta_{12}[k]]$ is the expected value w.r.t.all the measurement errors except the current one (i.e., $e_{12[0]}, e_{12[1]}, \cdots, e_{12[k-1]}$). Then $f_e(x)$ is the probability density function of $e_{12}[k]$. After some algebra and calculating the corresponding integrals, we obtain:

$$\mathbb{E}\left[\Delta_{12}[k+1]\right] = \mathbb{E}\left[\Delta_{12}[k]\right] \\ + k_{ef}\sigma\Phi^{-1}(\ell)\mathbb{E}\left[\Phi\left(-\Delta_{12}[k]/\sigma + \Phi^{-1}(\ell)\right)\right] \\ + k_{ef}\sigma\Phi^{-1}(\ell)\mathbb{E}\left[\Phi\left(-\Delta_{12}[k]/\sigma - \Phi^{-1}(\ell)\right)\right] \\ - k_{ef}\mathbb{E}\left[\Delta_{12}[k]\right] - k_{ef}\sigma\Phi^{-1}(\ell) \\ - k_{ef}\mathbb{E}\left[\Delta_{12}[k]\Phi\left(-\Delta_{12}[k]/\sigma + \Phi^{-1}(\ell)\right)\right] \\ + k_{ef}\mathbb{E}\left[\Delta_{12}[k]\Phi\left(-\Delta_{12}[k]/\sigma - \Phi^{-1}(\ell)\right)\right] \\ + \frac{\sigma k_{ef}}{\sqrt{2\pi}}\mathbb{E}\left[\exp\left(-(-\Delta_{12}[k] + \Phi(\ell))^2/2\sigma^2\right)\right] \\ - \frac{\sigma k_{ef}}{\sqrt{2\pi}}\mathbb{E}\left[\exp\left(-(-\Delta_{12}[k] - \Phi(\ell))^2/2\sigma^2\right)\right] \quad (100)$$

We approximate the nonlinear terms in the previous equation with their first-order Taylor series w.r.t. $\Delta_{12}[k]$ and centered

19

around $\Delta_{12}[k] = 0$. Thus, we obtain:

$$
\mathbb{E}\left[\Delta_{12}[k+1]\right] \approx -2k_{ef}\ell\mathbb{E}[\Delta_{12}[k]] + \mathbb{E}\left[\Delta_{12}[k]\right]
$$
$$
- k_{ef}\sqrt{\frac{2}{\pi}}\Phi^{-1}(\ell)\exp\left(-\frac{(\Phi^{-1}(\ell))^2}{2\sigma^2}\right)\mathbb{E}\left[\Delta_{12}[k]\right]
$$
$$
+ k_{ef}\sqrt{\frac{2}{\sigma^2\pi}}\Phi^{-1}(\ell)\exp\left(-\frac{(\Phi^{-1}(\ell))^2}{2\sigma^2}\right)\mathbb{E}\left[\Delta_{12}[k]\right] \quad (101)
$$
$$
= (1 - k'_{ef})\mathbb{E}\left[\Delta_{12}[k]\right]
$$

where

$$
k'_{ef} = k_{ef}\left(\left(1-\frac{1}{\sigma}\right)\sqrt{\frac{2}{\pi}}\Phi^{-1}(\ell)\exp\left(-\frac{(\Phi^{-1}(\ell))^2}{2\sigma^2}\right) + 2\ell\right). \quad (102)
$$

As long as $|1 - k'_{ef}| \in (0,1)$, the system (101) will be stable and we will have:

$$
\lim_{k\to\infty}\mathbb{E}\left[\Delta_{12}[k]\right] = 0 \quad (103)
$$

Nevertheless, if $k'_{ef} \in (0,1)$, then the system operates in an overdamped regime, but if $k'_{ef} \in (1,2)$, then it operates in an underdamped regime.

Calculating the exact variance of the system under its steady state involves considering extremely long equations and solving complicated integrals of highly nonlinear functions. Nevertheless, it is possible to analytically calculate the conditional variance of the system given $\Delta_{12}[k] = 0$, which corresponds to the expected value of the system in the steady state and it is given by:

$$
\text{var}[\Delta_{12}[k+1]|\Delta_{12}[k]{=}0] = 2k_{ef}^2\sigma^2\Phi^{-1}(\ell)\ell + 2k_{ef}^2\sigma^2\ell
$$
$$
+ 2k_{ef}^2\sigma^2\Phi^{-1}(\ell)/\sqrt{2\pi}\exp\left(-(\Phi^{-1}(\ell))^2/2\right) \quad (104)
$$

This equation was obtained by first setting $\Delta_{12}[k] = 0$ and following a similar procedure as that used to calculate the expected value in (100). The conditional variance (104) is a strictly increasing function of ℓ for $\ell \in [0, 0.5]$.

We refer to the duration T expressing the number of consecutive time steps in which $p_1[k]$ remains constant ($\forall o \in [1,T] : p_{1[k+o]} = p_1[k]$) as the coherence time k_{coh} of the system. This coherence time conditioned on $\Delta_{12}[k]$ is a random variable that follows a geometrical distribution

$$
\mathrm{P}(T = n \,|\, \Delta_{12}[k] = z) = (q_1[k](z))^{n-1}\bar{q}_1[k](z), \quad (105)
$$

and conditional expected value

$$
\mathbb{E}[T \,|\, \Delta_{12}[k] = z] = \frac{1}{\bar{q}_1[k](z)}. \quad (106)
$$

The unconditional distribution and expected values are:

$$
\mathrm{P}(T = n) = \int_{-\infty}^{+\infty}\mathrm{P}(T = n \,|\, \Delta_{12}[k] = z)f_{\Delta_{12}}[k](z)\mathrm{d}z,
$$
$$
\mathbb{E}[T] = \int_{-\infty}^{+\infty}\frac{f_{\Delta_{12}}[k](z)}{\bar{q}_1[k](z)}\mathrm{d}z, \quad (107)
$$

where $f_{\Delta_{12}}[k](z)$ is the probability density function of $\Delta_{12}[k]$ during the steady state. Analytically deriving such a probability density function is extremely complicated, but after performing a large numerical analysis on numerical simulations,

k_{ef}	0.1	0.5	1.0	1.5	1.9
$\beta(k_{ef})$	0.7251	0.8266	1.043	1.498	3.177

TABLE II: Table for $\beta(k_{ef})$.

ℓ	0.45	0.3	0.2	0.1	0.05
From simulations	1.1084	1.6491	2.4722	4.8897	9.6895
From (107), (108)	1.1083	1.6494	.4604	4.8912	9.7489

TABLE III: Expected coherence time k_{coh} ($\sigma_m = 0.1$, $k_{ef} = 0.1$).

we observed that 1) the steady-state probability distribution $f_{\Delta_{12}}[k](z)$ of the system (97) can be considered Gaussian for the range $\{\ell, k_{ef}\} \in [0.1, 0.5] \times [0.1, 1]$. This hypothesis is validated by the extremely low values of the Kullback-Leiber divergence calculated between the density function obtained through simulations and the Gaussian distribution, see Table IV; 2) the variance of $\Delta_{12}[k]$ decreases with ℓ; 3) for $k_{ef} \in [0.1, 1.9]$ and $\ell \in [0.01, 0.5]$, a good approximation for the variance of $\Delta_{12}[k]$ is[4]:

$$
\sigma_{ss,\text{res}}^2 = \frac{k_{ef}\sigma^2}{2 - k_{ef}}\exp\left(\beta(k_{ef})\Phi^{-1}(\ell)\right) \quad (108)
$$

where $\beta(k_{ef})$ is given in table II. We then observe in table III a comparison of the expected coherence time estimated by simulation and the numerical approximation.

D. 2 active agents

We consider again the same unidimensional scenario, except that now Agent 2 is also active and implements the same controller as Agent 1. This scenario corresponds to the real-world case when two agents are mutually observing. The dynamics of such agents is expressed as follows:

$$
p_1[k+1] = k_{ef}\text{clamp}(y_{12}[k], p_{12}[k]^m - p_{12}^d) + p_1[k],
$$
$$
p_2[k+1] = k_{ef}\text{clamp}(y_{21}[k], p_{21}[k]^m - p_{21}^d) + p_2[k],
$$
$$
y_{ij}[k] = \text{sign}\left(\delta_{ij}[k]\right)\hat{\sigma}_{ij}[k]\Phi^{-1}(\ell) + \delta_{ij}[k], \quad (109)
$$
$$
\delta_{ij}[k] = \Delta_{ij}[k] + e_{ij}[k],
$$
$$
\Delta_{ij}[k]^m = p_j[k] - p_i[k] - p_{ij}^d,
$$
$$
p_{ij}[k]^m = p_j[k] - p_i[k] + e_{ij}[k],
$$

where $e_{ij}[k]$ and $e_{ji}[k]$ are statistically independent, but we will assume for simplicity that they have the same variance. Thus, $p_{21}^d = -p_{12}^d$, although in general, $p_{21}[k]^m \neq -p_{12}[k]^m$. Next, we analyse the following system:

$$
\Delta_{12}[k+1] = k_{ef}\text{clamp}(y_{21}[k], p_{21}[k]^m - p_{21}^d) \quad (110)
$$
$$
- k_{ef}\text{clamp}(y_{12}[k], p_{12}[k]^m - p_{12}^d) + \Delta_{12}[k]
$$

Using similar reasoning as in the previous case for 1 Agent, we approximate system (110) with the following switched system

$$
\Delta_{12}[k+1] = \begin{cases} k_{ef}C_{12}[k] + \Delta_{12}[k+1] \\ \quad \text{if } |\Delta_{12}[k]'| \le -\Phi^{-1}(\ell) + 3, \\ k_{ef}(y_{21}[k] - y_{12}[k]) + \Delta_{12}[k+1] \text{ otw.,} \end{cases} \quad (111)
$$
$$
C_{12}[k] = \text{clamp}(y_{21}[k], p_{21}[k]^m - p_{21}^d)
$$
$$
- \text{clamp}(y_{12}[k], p_{12}[k]^m - p_{12}^d),
$$

	$k_{ef} = 0.1$	$k_{ef} = 0.5$	$k_{ef} = 1.0$
$\ell = 0.1$	0.0018	0.0035	0.0122
$\ell = 0.3$	0.0005	0.0009	0.0051
$\ell = 0.5$	0.0003	0.0002	0.0002

TABLE IV: Kullback-Leibler divergence between the $\Delta[k]$ probability density function measured from simulations p_S and $p_G(z) = 1/(\hat{\sigma}_m\sqrt{2\pi})\exp\left(-z^2/2\hat{\sigma}_m^2\right)$. $\hat{\sigma}_m$ is the variance of $\Delta[k]$ estimated from simulations.

[4]This approximation was obtained heuristically by extensive analysis of simulation data and its error was observed to be significantly low.

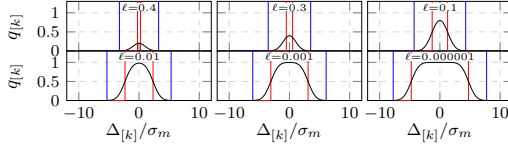## CHAPTER 4. RESEARCH MOTIVATED BY THE *UVDAR* SYSTEM

Fig. 19: Conditional stopping probability conditioned on the offset $\Delta[k]$ from the measured target position with measurement variance σ_m. The red line shows the symmetric restrained target position derived from ℓ, while the blue lines delimit the area into which the system state converges akin to proportional control.

with $\Delta_{12}[k]' = \Delta_{12}[k]/\sigma[k]$. For $\Delta_{12}[k]' \notin \mathcal{S}_\ell$ using (92), we obtain:

$$
\begin{aligned}
\mathbb{E}\left[\Delta_{12}[k+1]'\right] &\approx k_{ef}\Phi^{-1}(\ell)\left(\text{sign}\left(\Delta_{21}[k]'\right) - \text{sign}\left(\Delta_{12}[k]'\right)\right) \\
&\quad + (1 - k_{ef})\mathbb{E}\left[\Delta_{12}[k]'\right] + k_{ef}\mathbb{E}\left[\Delta_{21}[k]'\right], \\
&= -2k_{ef}\Phi^{-1}(\ell)\,\text{sign}\left(\Delta_{12}[k]'\right) \\
&\quad + (1 - 2k_{ef})\mathbb{E}\left[\Delta_{12}[k]'\right]
\end{aligned}
\tag{112}
$$

From (112), we observe that this system is stable for $|1 - 2k_{ef}| \leq 1$, i.e., for $k_{ef} \in [0,1]$. As opposed to the system (92) that is stable for $k_{ef} \in [0,2]$. This shows that as the number of agents interacting with each other increases, the range of the gain k_{ef} over which the system is stable shrinks.

Finally, if $k_{ef} \in [0,1]$, then the system (112) will enforce $\Delta_{12}[k] \in \mathcal{S}_\ell$ for some k. Once this happens, the probability that one agent has its input clamped rapidly increases, leading to the 1 mobile agent case where we have established that $\mathbb{E}[\Delta_{12}[k]] \to 0$. Eventually, the other agent may stop having its input clamped, but $\Delta_{12}[k]$ will be smaller, and the probability that any of the agents will have its input clamped will be even higher. This demonstrates the stability of the system.

REFERENCES

[1] 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 *ICUAS*, 2022, pp. 1264–1273.

[2] M. Coppola, K. N. McGuire, C. De Wagter, and G. C. H. E. de Croon, "A survey on swarming with micro air vehicles: Fundamental challenges and constraints," *Frontiers in Robotics and AI*, vol. 7, 2020.

[3] P. Stegagno, M. Cognetti, A. Franchi, and G. Oriolo, "Mutual localization using anonymous bearing measurements," in *IROS*, 2011.

[4] T. Chen, Q. Gao, and M. Guo, "An improved multiple UAVs cooperative flight algorithm based on leader follower strategy," in *CCDSC*, 2018.

[5] S. van der Helm, M. Coppola, K. N. McGuire, and G. C. H. E. de Croon, "On-board range-based relative localization for micro air vehicles in indoor leader–follower flight," *AuRo*, 2019.

[6] M. Vrba, D. Hert, and M. Saska, "Onboard marker-less detection and localization of non-cooperating drones for their safe interception by an autonomous aerial system," *RA-L*, vol. 4, no. 4, pp. 3402–3409, 2019.

[7] M. Vrba and M. Saska, "Marker-less micro aerial vehicle detection and localization using convolutional neural networks," *RA-L*, vol. 5, no. 2, pp. 2459–2466, 2020.

[8] J. Li, D. H. Ye, T. Chung, M. Kolsch, J. Wachs, and C. Bouman, "Multi-target detection and tracking from a single camera in Unmanned Aerial Vehicles (UAVs)," in *IROS*, 2016.

[9] A. Rozantsev, V. Lepetit, and P. Fua, "Flying objects detection from a single moving camera," in *CVPR*, 2015.

[10] K. R. Sapkota, S. Roelofsen, A. Rozantsev, V. Lepetit, D. Gillet, P. Fua, and A. Martinoli, "Vision-based Unmanned Aerial Vehicle detection and tracking for sense and avoid systems," in *IROS*, 2016.

[11] R. Opromolla, G. Fasano, and D. Accardo, "A vision-based approach to UAV detection and tracking in cooperative applications," *Sensors*, vol. 18, no. 10, 2018.

[12] M. Saqib, S. D. Khan, N. Sharma, and M. Blumenstein, "A study on detecting drones using deep convolutional neural networks," in *AVSS*, 2017.

[13] A. Schumann, L. Sommer, J. Klatte, T. Schuchert, and J. Beyerer, "Deep cross-domain flying object classification for robust UAV detection," in *AVSS*, 2017.

[14] 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 *CASE*, 2018.

[15] V. Walter, M. Saska, and A. Franchi, "Fast mutual relative localization of UAVs using ultraviolet LED markers," in *ICUAS*, 2018.

[16] V. Walter, N. Staub, A. Franchi, and M. Saska, "UVDAR system for visual relative localization with application to leader–follower formations of multirotor UAVs," *RA-L*, vol. 4, no. 3, pp. 2637–2644, 2019.

[17] V. Walter, M. Vrba, and M. Saska, "On training datasets for machine learning-based visual relative localization of micro-scale UAVs," in *ICRA*, 2020, pp. 10 674–10 680.

[18] J. Horyna, V. Walter, and M. Saska, "UVDAR-COM: UV-based relative localization of UAVs with integrated optical communication," in *ICUAS*, 2022.

[19] D. B. Licea, V. Walter, M. Ghogho, and M. Saska, "Optical communication-based identification for multi-UAV systems: Theory and practice," *arXiv preprint 2302.04770*, 2023.

[20] J. Cortés and M. Egerstedt, "Coordinated control of multi-robot systems: A survey," *SICE JCMSI*, vol. 10, no. 6, pp. 495–503, 2017.

[21] A. Franchi, C. Masone, V. Grabe, M. Ryll, H. H. Bülthoff, and P. R. Giordano, "Modeling and control of UAV bearing formations with bilateral high-level steering," *IJRR*, vol. 31, no. 12, 2012.

[22] F. Schiano, A. Franchi, D. Zelazo, and P. R. Giordano, "A rigidity-based decentralized bearing formation controller for groups of quadrotor UAVs," in *IROS*, 2016, pp. 5099–5106.

[23] D. Zelazo, A. Franchi, and P. R. Giordano, "Rigidity theory in SE(2) for unscaled relative position estimation using only bearing measurements," in *ECC*, 2014, pp. 2703–2708.

[24] K. Lu, S.-L. Dai, and X. Jin, "Fixed-time rigidity-based formation maneuvering for nonholonomic multirobot systems with prescribed performance," *Transactions on Cybernetics*, pp. 1–13, 2022.

[25] F. Zhang and N. E. Leonard, "Coordinated patterns of unit speed particles on a closed curve," *Systems & Control Letters*, vol. 56, no. 6, pp. 397–407, 2007.

[26] J. Lawton, R. Beard, and B. Young, "A decentralized approach to formation maneuvers," *Transactions on Robotics and Automation*, vol. 19, no. 6, pp. 933–941, 2003.

[27] F. Mehdifar, F. Hashemzadeh, M. Baradarannia, and M. de Queiroz, "Finite-time rigidity-based formation maneuvering of multiagent systems using distributed finite-time velocity estimators," *Transactions on Cybernetics*, vol. 49, no. 12, pp. 4473–4484, 2019.

[28] A. M. Amani, G. Chen, M. Jalili, X. Yu, and L. Stone, "Distributed rigidity recovery in distance-based formations using configuration lattice," *Transactions on Control of Network Systems*, vol. 7, no. 3, pp. 1547–1558, 2020.

[29] G. Michieletto, A. Cenedese, and A. Franchi, "Bearing rigidity theory in SE(3)," in *Conf. on Decision and Control*, 2016, pp. 5950–5955.

[30] J. Cortés, "Global and robust formation-shape stabilization of relative sensing networks," *Automatica*, vol. 45, no. 12, pp. 2754–2762, 2009.

[31] Y.-B. Bae, Y.-H. Lim, S.-M. Kang, and H.-S. Ahn, "Disturbance attenuation in distance-based formation control: A linear matrix inequality approach," in *CCTA*, 2018, pp. 1609–1614.

[32] Y.-B. Bae, Y.-H. Lim, and H.-S. Ahn, "Distributed robust adaptive gradient controller in distance-based formation control with exogenous disturbance," *Transactions on Automatic Control*, vol. 66, no. 6, pp. 2868–2874, 2021.

[33] L. Vasseur, O. Lecointe, J. Dento, N. Cherfaoui, V. Marion, and J. G. Morillon, "Leader-follower function for autonomous military convoys," in *Unmanned Ground Vehicle Technology*, vol. 5422, 2004, pp. 326 – 337.

[34] A. Gilbert, V. S. Chipade, and D. Panagou, "Robust leader-follower formation control for human-robot scenarios," in *ACC*, 2022, pp. 641–646.

[35] D. Van Vu, M. H. Trinh, P. D. Nguyen, and H.-S. Ahn, "Distance-based formation control with bounded disturbances," *Control Systems Letters*, vol. 5, no. 2, pp. 451–456, 2021.

[36] 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," *JIRS*, vol. 102, no. 26, pp. 1–28, 2021.

Chapter 5

Conclusion

This thesis has presented my work on the topic of mutual relative localization and stabilization of UAVs.

The first contribution of my work is to Vision-based Relative Localization (VRL) for UAVs, represented by the development of the *UVDAR* system, a unique smart sensor that became an enabling technology for real-world deployment of closely cooperating UAVs. This system is presented in this thesis in three publications detailing its core operational principles. The system was used extensively in research by our group, as outlined in the related sections of this thesis. Additionally, the system was used in cooperation with researchers from the TII institute [33]–[37], and it even shows early adoption by independent researchers [38].

The second contribution pertains to relative visual stabilization of UAVs, represented by development of vision-based fire extinguishing methods. These methods address the bidirectional interaction of the real UAV motion with the camera image in feedback. The merit of these methods is highlighted by their successful use in the fire-fighting challenge of the MBZIRC 2020 competition, as shown in the three publications presented in this thesis.

Lastly, I have contributed to the field of vision-based cooperation of UAVs in various modalities, as presented in four publications in the last section of this thesis. We have used *UVDAR* to create a practical annotated training dataset for aerial VRL of UAVs using Machine Learning (ML)-based detectors, which have seen a significant increase in popularity, and this dataset was published for use by the multi-robot community. Additionally, various cooperation methods were all tested in real deployment in outdoor conditions, which was enabled by the development of *UVDAR*. Besides being among the first to perform real outdoor sensor-based swarm flights, we have leveraged the novel features inherent in *UVDAR* to perform distributed formation flying. In particular, the last publication in this thesis proposes and shows a distributed formation-enforcing control (FEC) method in real-world deployment that takes into account the statistical distribution of the sensory noise inherent in a VRL system, which was not taken into consideration in prior literature. This method enables teams of UAVs to assume a formation, even in cases where the amount of noise in VRL would otherwise prevent it.

5.1 Future work

The primary direction in which I aim to continue my work is distributed formation control, as represented by my latest paper. Multiple specific aspects of this topic are yet to be explored, including a more in-depth analysis of the dynamics of a formation driven with noisy relative measurements, exploration of non-linear control action for greater optimality of the overall behavior, integration of collision-avoidance into formation-enforcing control, consideration of the non-Gaussian noise distribution of a visual relative localization, and more. Furthermore, the *UVDAR* system is still being improved by new technologies and ideas by myself and some of my younger colleagues, including non-linear signal trajectory extraction, usage of event-based cameras for significantly higher transmission band-width and tracking speed, and fusion with other sensors, such as UWB range sensors, for better 3D pose estimation precision.

Chapter 6

Author's publications

6.1 Thesis-related publications

Impact factors, citation counts from Web of Science (WoS), Scopus and Google Scholar (GS) and rankings were obtained on 8.1.2024. The two quantile ratings are obtained from Web of Science, according to the Journal Impact Factor and Journal Citation Indicator in the category relevant to the journal respectively. Conference ratings according to the Computing Research and Education Association of Australasia (CORE)[1] were added where applicable.

6.1.1 Thesis core publications

Core publications in peer-reviewed journals with impact factor

- **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, Jul. 2019, ISSN: 2377-3766. DOI: `10.1109/LRA.2019.2901683` **80% contribution, IF 5.9, Q1/Q2, Citations: 61 in WoS, 76 in Scopus, 129 in GS**
- V. Spurny, V. Pritzl, **V. Walter**, M. Petrlik, T. Baca, P. Stepan, D. Zaitlik, and M. Saska, "Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle," *IEEE Access*, vol. 9, pp. 15 872–15 890, Jan. 2021. DOI: `10.1109/ACCESS.2021.3052967` **15% contribution, IF 3.9, Q2, Citations: 26 in WoS, 29 in Scopus, 48 in GS**
- P. Petráček, **V. Walter**, T. Báča, 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, Dec. 2020. DOI: `https://doi.org/10.1088/1748-3190/abc6b3` **20% contribution, IF 3.8, Q3/Q3, Citations: 21 in WoS, 23 in Scopus, 37 in GS**

Core publications in conferences with CORE A or CORE A* rating

- **V. Walter**, M. Vrba, and M. Saska, "On training datasets for machine learning-based visual relative localization of micro-scale UAVs," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, © 2020 IEEE. Reprinted, with permission., Aug. 2020, pp. 10 674–10 680. DOI: `10.1109/ICRA40945.2020.9196947` **45% contribution, A*, Citations: 11 in Scopus, 28 in GS**

Core conference publications in WoS and Scopus

- **V. Walter**, M. Saska, and A. Franchi, "Fast mutual relative localization of uavs using ultraviolet led markers," in *2018 International Conference on Unmanned Aircraft*

---

[1] `http://portal.core.edu.au/conf-ranks/2051/`

System (ICUAS 2018), © 2018 IEEE. Reprinted, with permission., 2018 **82% contribution, Citations: 26 in WoS, 34 in Scopus, 91 in GS**

- **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 *14th IEEE International Conference on Automation Science and Engineering (CASE 2018)*, © 2018 IEEE. Reprinted, with permission., 2018 **73% contribution, Citations: 20 in WoS, 23 in Scopus, 62 in GS**
- **V. Walter**, V. Spurný, M. Petrlík, T. Báča, D. Žaitlík, and M. Saska, "Extinguishing of Ground Fires by Fully Autonomous UAVs Motivated by the MBZIRC 2020 Competition," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, © 2021 IEEE. Reprinted, with permission., Jun. 2021, pp. 787–793. DOI: `10.1109/ICUAS51884.2021.9476723` **60% contribution, Citations: 9 in Scopus, 13 in GS**

Other core publications

- **V. Walter**, V. Spurny, M. Petrlik, T. Báca, D. Zaitlík, L. Demkiv, and M. Saska, "Extinguishing real fires by fully autonomous multirotor UAVs in the MBZIRC 2020 competition," *Field Robotics*, vol. 2, pp. 406–436, Apr. 2022, ISSN: 2771-3989. DOI: `10.55417/fr.2022015` **15% contribution, Citations: 11 in GS** *- Note: Field Robotics is considered by the community to be successor to Journal of Field Robotics, impact factor is expected to be awarded in the upcoming year.*

Core publications under review

- D. B. Licea, **V. Walter**, M. Ghogho, and M. Saska, "Optical communication-based identification for multi-uav systems: Theory and practice," 2023. arXiv: `2302.04770` `[eess.SP]` **30% contribution, Submitted to Autonomous Robots**
- **V. Walter**, M. Vrba, D. B. Licea, and M. Saska, "Distributed formation-enforcing control for uavs robust to observation noise in relative pose measurements," 2023. arXiv: `2304.03057` `[cs.RO]` **40% contribution, Submitted to Transactions on Robotics**

6.1.2 Other thesis-related publications

- J. Horyna, **V. Walter**, and M. Saska, "UVDAR-COM: UV-Based Relative Localization of UAVs with Integrated Optical Communication," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, Jun. 2022. DOI: `10.1109/ICUAS54217.2022.9836151` **33% contribution, Citations: 2 in WoS, 4 in Scopus, 7 in GS**
- 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*, pp. 1–17, 2022. DOI: `https://doi.org/10.1007/s10514-022-10066-5` **15% contribution, IF 4.6, Q2/Q3, Citations: 5 in WoS, 6 in Scopus, 11 in GS**
- 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 *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE,

Jun. 2021, pp. 570–576. DOI: `10.1109/ICRA48506.2021.9561284` **7% contribution, A*, Citations: 10 in WoS, 14 in Scopus, 26 in GS**

- F. Novák, **V. Walter**, P. Petráček, T. Báča, and M. Saska, "Fast collective evasion in self-localized swarms of unmanned aerial vehicles," *Bioinspiration & Biomimetics*, vol. 16, no. 6, p. 066 025, Nov. 2021. DOI: `10.1088/1748-3190/ac3060` [Online]. Available: `https://doi.org/10.1088/1748-3190/ac3060` **5% contribution, IF 3.8, Q2/Q3, Citations: 7 in WoS, 7 in Scopus, 12 in GS**

- K. Jindal, A. Wang, D. Thakur, A. Zhou, V. Spurny, **V. Walter**, G. Broughton, T. Krajnik, M. Saska, and G. Loianno, "Design and deployment of an autonomous unmanned ground vehicle for urban firefighting scenarios," *Field Robotics*, vol. 1, pp. 186–202, 2022, ISSN: 2771-3989. DOI: `10.55417/fr.2022015` **equal contribution of authors**

- M. Vrba, V. Walter, V. Pritzl, M. Pliska, T. Báča, V. Spurný, D. Heřt, and M. Saska, "On onboard lidar-based flying object detection," 2023. arXiv: `2303.05404 [cs.RO]` **10% contribution, Submitted to Transactions on Robotics**

6.2 Other publications

- 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 *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, Jun. 2022, pp. 1264–1273. DOI: `10.1109/ICUAS54217.2022.9836083` **equal contribution of authors, Citations: 5 in WoS, 6 in Scopus, 25 in GS**

- **V. Walter**, T. Novák, and M. Saska, "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, 2018 **76% contribution, Citations: 4 in WoS, 6 in Scopus, 13 in GS**

Chapter 7

Citations of author's work

Citations were extracted from Web of Science where possible, with Scopus and Google Scholar being used where Web of Science entry was not available. Self-citations by myself or by my co-authors have been omitted.

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, Jul. 2019, ISSN: 2377-3766. DOI: 10.1109/LRA.2019.2901683; Citations according to Web of Science, accessed 4.12.2023.

- X. Cai, X. Zhu, and W. Yao, "Distributed adaptive time-varying formation of multi-uav systems under undirected graph," *IET INTELLIGENT TRANSPORT SYSTEMS*, 2023 NOV 2 2023, ISSN: 1751-956X. DOI: 10.1049/itr2.12446.

- 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, Nov. 2023, ISSN: 1270-9638. DOI: 10.1016/j.ast.2023.108665.

- H. Zhang, C. Miao, L. Zhang, Y. Zhang, Y. Li, and K. Fang, "A real-time simulator for navigation in gnss-denied environments of uav swarms," *APPLIED SCIENCES-BASEL*, vol. 13, no. 20, Oct. 2023. DOI: 10.3390/app132011278.

- R. Ming, R. Jiang, H. Luo, T. Lai, E. Guo, and Z. Zhou, "Comparative analysis of different uav swarm control methods on unmanned farms," *AGRONOMY-BASEL*, vol. 13, no. 10, Oct. 2023. DOI: 10.3390/agronomy13102499.

- Z. Ning, Y. Zhang, and S. Zhao, "Comparison of different pseudo-linear estimators for vision-based target motion estimation," *CONTROL THEORY AND TECHNOLOGY*, vol. 21, no. 3, pp. 448–457, Aug. 2023, ISSN: 2095-6983. DOI: 10.1007/s11768-023-00161-y.

- A. Merzi, V. Gazi, G. Fedele, and L. D'Alfonso, "Target capturing in an ellipsoidal region with a swarm of quadcopter agents with different social value orientation parameters," *JOURNAL OF INTELLIGENT & ROBOTIC SYSTEMS*, vol. 108, no. 4, Aug. 2023, ISSN: 0921-0296. DOI: 10.1007/s10846-023-01910-6.

- S. Huang, T. Wang, Y. Tang, Y. Hu, G. Xin, and D. Zhou, "Distributed and scalable cooperative formation of unmanned ground vehicles using deep reinforcement learning," *AEROSPACE*, vol. 10, no. 2, Feb. 2023. DOI: 10.3390/aerospace10020096.

- O. A. Hay, M. Chehadeh, A. Ayyad, M. Wahbah, M. A. Humais, I. Boiko, L. Seneviratne, and Y. Zweiri, "Noise-tolerant identification and tuning approach using deep neural networks for visual servoing applications," *IEEE TRANSACTIONS ON ROBOTICS*, vol. 39, no. 3, pp. 2276–2288, Jun. 2023, ISSN: 1552-3098. DOI: 10.1109/TRO.2023.3235586.

- Y. Li, P. Zhang, Z. Mei, and W. Dong, "Efficiently range-coupled position estimation for an aerial swarm with the confidence evaluation of onboard sensing," *IEEE-ASME TRANSACTIONS ON MECHATRONICS*, vol. 28, no. 4, pp. 2178–2188, Aug. 2023, ISSN: 1083-4435. DOI: 10.1109/TMECH.2022.3232796.

- G. A. Amran, S. Wang, M. A. A. Al-Qaness, S. A. H. Mohsan, R. Abbas, E. Ghaleb, S. Al-shathri, and M. A. Elaziz, "Efficient and secure wifi signal booster via unmanned aerial vehicleswifi repeater based on intelligence based localization swarm and blockchain," *MICROMACHINES*, vol. 13, no. 11, Nov. 2022. DOI: `10.3390/mi13111924`.

- H. Stuckey, A. Al-Radaideh, L. Sun, and W. Tang, "A spatial localization and attitude estimation system for unmanned aerial vehicles using a single dynamic vision sensor," *IEEE SENSORS JOURNAL*, vol. 22, no. 15, pp. 15 497–15 507, Aug. 2022, ISSN: 1530-437X. DOI: `10.1109/JSEN.2022.3187423`.

- 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, Dec. 2022, ISSN: 1552-3098. DOI: `10.1109/TRO.2022.3182503`.

- P. Zhang, G. Chen, Y. Li, and W. Dong, "Agile formation control of drone flocking enhanced with active vision-based relative localization," *IEEE ROBOTICS AND AUTOMATION LETTERS*, vol. 7, no. 3, pp. 6359–6366, Jul. 2022, ISSN: 2377-3766. DOI: `10.1109/LRA.2022.3171096`.

- M. H. Nordin, S. Sharma, A. Khan, M. Gianni, S. Rajendran, and R. Sutton, "Collaborative unmanned vehicles for inspection, maintenance, and repairs of offshore wind turbines," *DRONES*, vol. 6, no. 6, Jun. 2022. DOI: `10.3390/drones6060137`.

- Q. Ouyang, Z. Wu, Y. Cong, and Z. Wang, "Formation control of unmanned aerial vehicle swarms: A comprehensive review," *ASIAN JOURNAL OF CONTROL*, vol. 25, no. 1, pp. 570–593, Jan. 2023, ISSN: 1561-8625. DOI: `10.1002/asjc.2806`.

- S. Zheng, Z. Li, Y. Yin, Y. Liu, H. Zhang, P. Zheng, and X. Zou, "Multi-robot relative positioning and orientation system based on uwb range and graph optimization," *MEASUREMENT*, vol. 195, May 2022, ISSN: 0263-2241. DOI: `10.1016/j.measurement.2022.111068`.

- R. Ming, Z. Zhou, Z. Lyu, X. Luo, L. Zi, C. Song, Y. Zang, W. Liu, and R. Jiang, "Laser tracking leader-follower automatic cooperative navigation system for uavs," *INTERNATIONAL JOURNAL OF AGRICULTURAL AND BIOLOGICAL ENGINEERING*, vol. 15, no. 2, pp. 165–176, Mar. 2022, ISSN: 1934-6344. DOI: `10.25165/j.ijabe.20221502.6350`.

- Y. Jia, Y. Gao, H. Wang, and M. Chen, "A distributed method to form uav swarm based on moncular vision," in *2022 IEEE 28TH INTERNATIONAL CONFERENCE ON PARALLEL AND DISTRIBUTED SYSTEMS, ICPADS*, ser. International Conference on Parallel and Distributed Systems - Proceedings, IEEE 28th International Conference on Parallel and Distributed Systems (IEEE ICPADS), Nanjing, PEOPLES R CHINA, JAN 10-12, 2023, 2022, pp. 41–48, ISBN: 978-1-6654-7315-6. DOI: `10.1109/ICPADS56603.2022.00014`.

- 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 *2022 IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS AND SYSTEMS (IROS)*, ser. IEEE International Conference on Intelligent Robots and Systems, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, JAPAN, OCT 23-27, 2022, 2022, pp. 380–387, ISBN: 978-1-6654-7927-1. DOI: `10.1109/IROS47612.2022.9981115`.

- J. P. Queralta, Q. Li, F. Schiano, and T. Westerlund, "Vio-uwb-based collaborative localization and dense scene reconstruction within heterogeneous multi-robot systems," in *2022 INTERNATIONAL CONFERENCE ON ADVANCED ROBOTICS AND MECHATRONICS (ICARM 2022)*, 7th IEEE International Conference on Advanced Robotics and Mechatronics, Guilin, PEOPLES R CHINA, JUL 09-11, 2022, 2022, pp. 87–94, ISBN: 978-1-6654-8306-3. DOI: `10.1109/ICARM54641.2022.9959470`.

- A. Suarez, R. Salmoral, A. Garofano-Soldado, G. Heredia, and A. Ollero, "Aerial device delivery for power line inspection and maintenance," in *2022 INTERNATIONAL CONFERENCE ON UN-MANNED AIRCRAFT SYSTEMS (ICUAS)*, ser. International Conference on Unmanned Aircraft Systems, International Conference on Unmanned Aircraft Systems (ICUAS), Dubrovnik, CROA-TIA, JUN 21-24, 2022, 2022, pp. 30–38, ISBN: 978-1-6654-0593-5. DOI: 10.1109/ICUAS54217.2022.9836039.

- M. A. Merzi, V. Gazi, G. Fedele, L. D'Alfonso, and A. Bono, "Target capturing in an ellipsoidal region with a swarm of quadrotor uavs," in *2022 INTERNATIONAL CONFERENCE ON UN-MANNED AIRCRAFT SYSTEMS (ICUAS)*, ser. International Conference on Unmanned Aircraft Systems, International Conference on Unmanned Aircraft Systems (ICUAS), Dubrovnik, CROA-TIA, JUN 21-24, 2022, 2022, pp. 972–981, ISBN: 978-1-6654-0593-5. DOI: 10.1109/ICUAS54217.2022.9836166.

- H. Shen, Q. Zong, H. Lu, X. Zhang, B. Tian, and L. He, "A distributed approach for lidar-based relative state estimation of multi-uav in gps-denied environments," *CHINESE JOURNAL OF AERONAUTICS*, vol. 35, no. 1, pp. 59–69, Jan. 2022, ISSN: 1000-9361. DOI: 10.1016/j.cja.2021.04.021.

- 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, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2022.3158758.

- K. M. Kabore and S. Guler, "Distributed formation control of drones with onboard perception," *IEEE-ASME TRANSACTIONS ON MECHATRONICS*, vol. 27, no. 5, pp. 3121–3131, Oct. 2022, ISSN: 1083-4435. DOI: 10.1109/TMECH.2021.3110660.

- T. Elmokadem and A. V. Savkin, "Towards fully autonomous uavs: A survey," *SENSORS*, vol. 21, no. 18, Sep. 2021. DOI: 10.3390/s21186223.

- R. Ming, Z. Zhou, X. Luo, W. Liu, Z. Le, C. Song, R. Jiang, and Y. Zang, "Optical tracking system for multi-uav clustering," *IEEE SENSORS JOURNAL*, vol. 21, no. 17, pp. 19 382–19 394, Sep. 2021, ISSN: 1530-437X. DOI: 10.1109/JSEN.2021.3091280.

- T. Tazrin, M. M. Fouda, Z. M. Fadlullah, and N. Nasser, "Uv-cds: An energy-efficient scheduling of uavs for premises sterilization," *IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING*, vol. 5, no. 3, pp. 1191–1201, Sep. 2021, ISSN: 2473-2400. DOI: 10.1109/TGCN.2021.3074536.

- F. Schilling, F. Schiano, and D. Floreano, "Vision-based drone flocking in outdoor environments," *IEEE ROBOTICS AND AUTOMATION LETTERS*, vol. 6, no. 2, pp. 2954–2961, Apr. 2021, ISSN: 2377-3766. DOI: 10.1109/LRA.2021.3062298.

- T. Ziegler, M. Karrer, P. Schmuck, and M. Chli, "Distributed formation estimation via pair-wise distance measurements," *IEEE ROBOTICS AND AUTOMATION LETTERS*, vol. 6, no. 2, pp. 3017–3024, Apr. 2021, ISSN: 2377-3766. DOI: 10.1109/LRA.2021.3062347.

- M. Pavliv, F. Schiano, C. Reardon, D. Floreano, and G. Loianno, "Tracking and relative localization of drone swarms with a vision-based headset," *IEEE ROBOTICS AND AUTOMATION LET-TERS*, vol. 6, no. 2, pp. 1455–1462, Apr. 2021, ISSN: 2377-3766. DOI: 10.1109/LRA.2021.3051565.

- H. Janjua, F. Yang, M. Ammar, D. Newton, S. Ro, S. Michiels, and D. Hughes, "Oslo: Optical sensor localization through mesh networked cameras," in *2021 IEEE 22ND INTERNATIONAL SYMPOSIUM ON A WORLD OF WIRELESS, MOBILE AND MULTIMEDIA NETWORKS (WOWMOM 2021)*, ser. IEEE International Symposium on a World of Wireless, Mobile and Mul-timedia Networks, 22nd IEEE International Symposium on a World of Wireless, Mobile and Multi-media Networks (IEEE WoWMoM), ELECTR NETWORK, JUN 07-11, 2021, 2021, pp. 139–148, ISBN: 978-1-6654-2263-5. DOI: 10.1109/WoWMoM51794.2021.00027.

- L. Qingqing, Y. Xianjia, J. P. Queralta, and T. Westerlund, "Adaptive lidar scan frame integration: Tracking known mavs in 3d point clouds," in *2021 20TH INTERNATIONAL CONFERENCE ON ADVANCED ROBOTICS (ICAR)*, 20th International Conference on Advanced Robotics (ICAR), ELECTR NETWORK, DEC 07-10, 2021, 2021, pp. 1079–1086, ISBN: 978-1-6654-3684-7. DOI: 10.1109/ICAR53236.2021.9659483.

- H. Stuckey, A. Al-Radaideh, L. Escamilla, L. Sun, L. G. Carrillo, and W. Tang, "An optical spatial localization system for tracking unmanned aerial vehicles using a single dynamic vision sensor," in *2021 IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS AND SYSTEMS (IROS)*, ser. IEEE International Conference on Intelligent Robots and Systems, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), ELECTR NETWORK, SEP 27-OCT 01, 2021, 2021, pp. 3093–3100, ISBN: 978-1-6654-1714-3. DOI: 10.1109/IROS51168.2021.9636665.

- N. J. Sanket, C. D. Singh, C. M. Parameshwara, C. Fermueller, G. C. H. E. de Croon, and Y. Aloimonos, "Evpropnet: Detecting drones by finding propellers for mid-air landing and following," in *ROBOTICS: SCIENCE AND SYSTEM XVII*, D. Shell, M. Toussaint, and M. Hsieh, Eds., ser. Robotics - Science and Systems, Conference on Robotics - Science and Systems, ELECTR NETWORK, JUL 12-16, 2021, 2021, ISBN: 978-0-9923747-7-8.

V. Spurny, V. Pritzl, V. Walter, M. Petrlik, T. Baca, P. Stepan, D. Zaitlik, and M. Saska, "Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle," *IEEE Access*, vol. 9, pp. 15 872–15 890, Jan. 2021. DOI: 10.1109/ACCESS.2021.3052967; Citations according to Web of Science, accessed 4.12.2023.

- C. Son, "Disaster ergonomics: A human factors approach to address escalating challenges from disasters," *COGNITION TECHNOLOGY & WORK*, vol. 25, no. 4, pp. 325–344, Nov. 2023, ISSN: 1435-5558. DOI: 10.1007/s10111-023-00736-4.

- D. Perez-Saura, M. Fernandez-Cortizas, R. Perez-Segui, P. Arias-Perez, and P. Campoy, "Urban firefighting drones: Precise throwing from uav," *JOURNAL OF INTELLIGENT & ROBOTIC SYSTEMS*, vol. 108, no. 4, Aug. 2023, ISSN: 0921-0296. DOI: 10.1007/s10846-023-01883-6.

- T. Huynh and Y.-B. Kim, "Motion control system design for a novel water-powered aerial system for firefighting with flow-regulating actuators," *DRONES*, vol. 7, no. 3, Mar. 2023. DOI: 10.3390/drones7030162.

- T. Huynh, D.-H. Lee, and Y.-B. Kim, "Study on actuator performance evaluation of aerial water-powered system for firefighting applications," *APPLIED SCIENCES-BASEL*, vol. 13, no. 3, Feb. 2023. DOI: 10.3390/app13031965.

- N. F. Talavera, J. J. Roldan-Gomez, F. Martin, and M. C. Rodriguez-Sanchez, "An autonomous ground robot to support firefighters' interventions in indoor emergencies," *JOURNAL OF FIELD ROBOTICS*, vol. 40, no. 3, pp. 614–625, May 2023, ISSN: 1556-4959. DOI: 10.1002/rob.22150.

- T.-L. Xiao, Q. Yang, X.-D. Gao, Z.-Y. Lu, Y.-Y. Ma, S.-W. Jeon, and J. Zhang, "Variation encoded large-scale swarm optimizers for path planning of unmanned aerial vehicle," in *PROCEEDINGS OF THE 2023 GENETIC AND EVOLUTIONARY COMPUTATION CONFERENCE, GECCO 2023*, L. Paquete, Ed., Genetic and Evolutionary Computation Conference (GECCO), Lisbon, PORTUGAL, JUL 15-19, 2023, 2023, pp. 102–110, ISBN: 979-8-4007-0119-1. DOI: 10.1145/3583131.3590357.

- T.-V. Nguyen, N. P. Nguyen, C. Kim, and N.-N. Dao, "Intelligent aerial video streaming: Achievements and challenges," *JOURNAL OF NETWORK AND COMPUTER APPLICATIONS*, vol. 211, Feb. 2023, ISSN: 1084-8045. DOI: 10.1016/j.jnca.2022.103564.

- A. Hemmati and A. M. Rahmani, "The internet of autonomous things applications: A taxonomy, technologies, and future directions," *INTERNET OF THINGS*, vol. 20, Nov. 2022, ISSN: 2543-1536. DOI: `10.1016/j.iot.2022.100635`.

- J. Balen, H. Glavas, K. Vdovjak, and J. Jakab, "Obtaining infrared thermal camera sensor calibration data for implementation in firebot autonomous fire protection robot system," *APPLIED SCIENCES-BASEL*, vol. 12, no. 22, Nov. 2022. DOI: `10.3390/app122211657`.

- N. A. K. Kaimkhani, M. Noman, S. Rahim, and H. Bin Liaqat, "Uav with vision to recognise vehicle number plates," *MOBILE INFORMATION SYSTEMS*, vol. 2022, Oct. 2022, ISSN: 1574-017X. DOI: `10.1155/2022/7655833`.

- I. L. H. Alsammak, M. A. Mahmoud, H. Aris, M. AlKilabi, and M. N. Mahdi, "The use of swarms of unmanned aerial vehicles in mitigating area coverage challenges of forest-fire-extinguishing activities: A systematic literature review," *FORESTS*, vol. 13, no. 5, May 2022. DOI: `10.3390/f13050811`.

- M. Hu, K. Lee, H. Ahn, A. Choi, H. Kim, and K. You, "Stabilization and tracking of a quadrotor using modified sigmoid sliding mode control," *SENSORS*, vol. 22, no. 10, May 2022. DOI: `10.3390/s22103618`.

- W. Chen, Z. Lin, P. Peng, and W. Dong, "Ctff: An efficient strategy for continuously targeting fire fighting with quadrotor," in *2022 INTERNATIONAL CONFERENCE ON ADVANCED ROBOTICS AND MECHATRONICS (ICARM 2022)*, 7th IEEE International Conference on Advanced Robotics and Mechatronics, Guilin, PEOPLES R CHINA, JUL 09-11, 2022, 2022, pp. 163–168, ISBN: 978-1-6654-8306-3. DOI: `10.1109/ICARM54641.2022.9959547`.

- A. Farooq, A. Anastasiou, N. Souli, C. Laoudias, P. S. Kolios, and T. Theocharides, "Uav autonomous indoor exploration and mapping for sar missions: Reflections from the icuas 2022 competition," in *2022 19TH INTERNATIONAL CONFERENCE ON UBIQUITOUS ROBOTS (UR)*, 19th International Conference on Ubiquitous Robots (UR), Jeju, SOUTH KOREA, JUL 04-06, 2022, 2022, pp. 621–626, ISBN: 978-1-6654-8253-0. DOI: `10.1109/UR55393.2022.9866527`.

- W. Zhuo, Z. Zhao, K. H. Chiu, S. Li, S. Ha, C.-H. Lee, and S.-H. G. Chan, "Grafics: Graph embedding-based floor identification using crowdsourced rf signals," in *2022 IEEE 42ND INTERNATIONAL CONFERENCE ON DISTRIBUTED COMPUTING SYSTEMS (ICDCS 2022)*, ser. IEEE International Conference on Distributed Computing Systems, 42nd IEEE International Conference on Distributed Computing Systems (ICDCS), Bologna, ITALY, JUL 10-13, 2022, 2022, pp. 1051–1061, ISBN: 978-1-6654-7177-0. DOI: `10.1109/ICDCS54860.2022.00105`.

- A. Farooq, C. Laoudias, P. S. Kolios, and T. Theocharides, "Quantitative and qualitative assessment of indoor exploration algorithms for autonomous uavs," in *2022 INTERNATIONAL CONFERENCE ON UNMANNED AIRCRAFT SYSTEMS (ICUAS)*, ser. International Conference on Unmanned Aircraft Systems, International Conference on Unmanned Aircraft Systems (ICUAS), Dubrovnik, CROATIA, JUN 21-24, 2022, 2022, pp. 487–496, ISBN: 978-1-6654-0593-5. DOI: `10.1109/ICUAS54217.2022.9836079`.

- P. M. Jacob, J. Moni, R. B. Robins, M. E. Varghese, S. S. Babu, and V. K. Bose, "An intelligent fire detection and extinguishing assistant system using internet of things (iot)," in *2022 INTERNATIONAL CONFERENCE ON DECISION AID SCIENCES AND APPLICATIONS (DASA)*, International Conference on Decision Aid Sciences and Applications (DASA), Chiangrai, THAILAND, MAR 23-25, 2022, 2022, pp. 1057–1061, ISBN: 978-1-6654-9501-1. DOI: `10.1109/DASA54658.2022.9765126`.

- A. Ivanovic and M. Orsag, "Parabolic airdrop trajectory planning for multirotor unmanned aerial vehicles," *IEEE ACCESS*, vol. 10, pp. 36 907–36 923, 2022, ISSN: 2169-3536. DOI: `10.1109/ACCESS.2022.3164434`.

- G. Kuznetsov, N. Kopylov, E. Sushkina, and A. Zhdanova, "Adaptation of fire-fighting systems to localization of fires in the premises: Review," *ENERGIES*, vol. 15, no. 2, Jan. 2022. DOI: `10.3390/en15020522`.

- D.-H. Lee, T. Huynh, Y.-B. Kim, and C. Soumayya, "Motion control system design for a flying-type firefighting system with water jet actuators," *ACTUATORS*, vol. 10, no. 10, Oct. 2021. DOI: `10.3390/act10100275`.

P. Petráček, V. Walter, T. Báča, 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, Dec. 2020. DOI: `https://doi.org/10.1088/1748-3190/abc6b3`; Citations according to Web of Science, accessed 4.12.2023.

- H. Xiong, Y. Ding, and J. Liu, "Compact and ordered swarms of unmanned aerial vehicles in cluttered environments," *BIOINSPIRATION & BIOMIMETICS*, vol. 18, no. 5, Sep. 2023, ISSN: 1748-3182. DOI: `10.1088/1748-3190/aced76`.

- 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, May 2023, ISSN: 1270-9638. DOI: `10.1016/j.ast.2023.108218`.

- 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, Mar. 2023, ISSN: 2377-3766. DOI: `10.1109/LRA.2023.3240376`.

- X. Wang, F. Wang, Z. Nie, Y. Ai, and T. Hu, "Optiswarm: Optical swarm robots using implicit cooperation," *IEEE SENSORS JOURNAL*, vol. 22, no. 24, pp. 24 380–24 394, Dec. 2022, ISSN: 1530-437X. DOI: `10.1109/JSEN.2022.3216445`.

- 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, Jun. 2022, ISSN: 2158-107X.

- E. Soria, "Swarms of flying robots in unknown environments," *SCIENCE ROBOTICS*, vol. 7, no. 66, May 2022, ISSN: 2470-9476. DOI: `10.1126/scirobotics.abq2215`.

- R. Ourari, K. Cui, A. Elshamanhory, and H. Koeppl, "Nearest-neighbor-based collision avoidance for quadrotors via reinforcement learning," in *2022 IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION (ICRA 2022)*, IEEE International Conference on Robotics and Automation (ICRA), Philadelphia, PA, MAY 23-27, 2022, 2022, ISBN: 978-1-7281-9681-7. DOI: `10.1109/ICRA.46639.2022.9812221`.

- 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, ISSN: 1079-8587. DOI: `10.32604/iasc.2022.024577`.

- 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, ISSN: 2169-3536. DOI: `10.1109/ACCESS.2022.3158758`.

- 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, Jan. 2022, ISSN: 2377-3766. DOI: `10.1109/LRA.2021.3118091`.

- Z. Sun, "Robot swarm navigation: Methods, analysis, and applications," in *2021 2ND INTERNATIONAL CONFERENCE ON BIG DATA & ARTIFICIAL INTELLIGENCE & SOFTWARE ENGINEERING (ICBASE 2021)*, 2nd International Conference on Big Data and Artificial Intelligence and Software Engineering (ICBASE), Zhuhai, PEOPLES R CHINA, SEP 24-26, 2021, 2021, pp. 666+, ISBN: 978-1-6654-2709-8. DOI: 10.1109/ICBASE53849.2021.00131.

> V. Walter, M. Vrba, and M. Saska, "On training datasets for machine learning-based visual relative localization of micro-scale UAVs," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, © 2020 IEEE. Reprinted, with permission., Aug. 2020, pp. 10 674–10 680. DOI: 10.1109/ICRA40945.2020.9196947; Citations according to Scopus, accessed 4.12.2023.

- Y. Zhao, Z. Ju, T. Sun, F. Dong, J. Li, R. Yang, Q. Fu, C. Lian, and P. Shan, "Tgc-yolov5: An enhanced yolov5 drone detection model based on transformer, gam & ca attention mechanism," *Drones*, vol. 7, no. 7, 2023, cited By 1. DOI: 10.3390/drones7070446
[Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85166305221&doi=10.3390%2fdrones7070446&partnerID=40&md5=6e27a36ef24c411c78c946e603af9bfb.

- F. Causa, R. Opromolla, and G. Fasano, "Closed loop integration of air-to-air visual measurements for cooperative uav navigation in gnss challenging environments," *Aerospace Science and Technology*, vol. 130, 2022, cited By 2. DOI: 10.1016/j.ast.2022.107947
[Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85140440544&doi=10.1016%2fj.ast.2022.107947&partnerID=40&md5=4d80838e93de3d846ed63ca43e74f1a3.

- C. Wang, Y. Su, J. Wang, T. Wang, and Q. Gao, "Uavswarm dataset: An unmanned aerial vehicle swarm dataset for multiple object tracking," *Remote Sensing*, vol. 14, no. 11, 2022, cited By 5. DOI: 10.3390/rs14112601
[Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85131377193&doi=10.3390%2frs14112601&partnerID=40&md5=84279e0ca2bee7b0a83114f2a4643fd5.

- S. Li, C. De Wagter, and G. De Croon, "Self-supervised monocular multi-robot relative localization with efficient deep neural networks," cited By 5, 2022, pp. 9689–9695. DOI: 10.1109/ICRA46639.2022.9812150
[Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85136328367&doi=10.1109%2fICRA46639.2022.9812150&partnerID=40&md5=5042f8d42730936d9c0ae4c14de1e2a0.

- F. Causa, R. Opromolla, and G. Fasano, "Cooperative navigation and visual tracking with passive ranging for uav flight in gnss-challenging environments," cited By 7, 2021, pp. 1538–1547. DOI: 10.1109/ICUAS51884.2021.9476681
[Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85111466113&doi=10.1109%2fICUAS51884.2021.9476681&partnerID=40&md5=e556f305b0e39e4b0d413bb4306584cc.

- Y. Zheng, Z. Chen, D. Lv, Z. Li, Z. Lan, and S. Zhao, "Air-to-air visual detection of micro-uavs: An experimental evaluation of deep learning," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1020–1027, 2021, cited By 29. DOI: 10.1109/LRA.2021.3056059
[Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85100705092&doi=10.1109%2fLRA.2021.3056059&partnerID=40&md5=4074331072bcd0a1bd4bd3d913474c36.

- F. Dadboud, V. Patel, V. Mehta, M. Bolic, and I. Mantegh, "Single-stage uav detection and classification with yolov5: Mosaic data augmentation and panet," cited By 24, 2021. DOI: 10.1109/AVSS52988.2021.9663841
[Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85124967820&doi=10.1109%2fAVSS52988.2021.9663841&partnerID=40&md5=a17d0333963b3ed3fd43504dfce21237.

V. Walter, M. Saska, and A. Franchi, "Fast mutual relative localization of uavs using ultraviolet led markers," in *2018 International Conference on Unmanned Aircraft System (ICUAS 2018)*, © 2018 IEEE. Reprinted, with permission., 2018; Citations according to Web of Science, accessed 4.12.2023.

- E. Omia, H. Bae, E. Park, M. S. Kim, I. Baek, I. Kabenge, and B.-K. Cho, "Remote sensing in field crop monitoring: A comprehensive review of sensor systems, data analyses and recent advances," *REMOTE SENSING*, vol. 15, no. 2, Jan. 2023. DOI: 10.3390/rs15020354.

- G. A. Amran, S. Wang, M. A. A. Al-Qaness, S. A. H. Mohsan, R. Abbas, E. Ghaleb, S. Al-shathri, and M. A. Elaziz, "Efficient and secure wifi signal booster via unmanned aerial vehicleswifi repeater based on intelligence based localization swarm and blockchain," *MICROMACHINES*, vol. 13, no. 11, Nov. 2022. DOI: 10.3390/mi13111924.

- P. Zhang, G. Chen, Y. Li, and W. Dong, "Agile formation control of drone flocking enhanced with active vision-based relative localization," *IEEE ROBOTICS AND AUTOMATION LETTERS*, vol. 7, no. 3, pp. 6359–6366, Jul. 2022, ISSN: 2377-3766. DOI: 10.1109/LRA.2022.3171096.

- H. Jafarbiglu and A. Pourreza, "A comprehensive review of remote sensing platforms, sensors, and applications in nut crops," *COMPUTERS AND ELECTRONICS IN AGRICULTURE*, vol. 197, Jun. 2022, ISSN: 0168-1699. DOI: 10.1016/j.compag.2022.106844.

- Z. Wang, Y. Ng, J. Henderson, and R. Mahony, "Smart visual beacons with asynchronous optical communications using event cameras," in *2022 IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS AND SYSTEMS (IROS)*, ser. IEEE International Conference on Intelligent Robots and Systems, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, JAPAN, OCT 23-27, 2022, 2022, pp. 3793–3799, ISBN: 978-1-6654-7927-1. DOI: 10.1109/IROS47612.2022.9982016.

- J. P. Queralta, Q. Li, F. Schiano, and T. Westerlund, "Vio-uwb-based collaborative localization and dense scene reconstruction within heterogeneous multi-robot systems," in *2022 INTERNATIONAL CONFERENCE ON ADVANCED ROBOTICS AND MECHATRONICS (ICARM 2022)*, 7th IEEE International Conference on Advanced Robotics and Mechatronics, Guilin, PEOPLES R CHINA, JUL 09-11, 2022, 2022, pp. 87–94, ISBN: 978-1-6654-8306-3. DOI: 10.1109/ICARM54641.2022.9959470.

- K. M. Kabore and S. Guler, "Practical formation acquisition mechanism for nonholonomic leader-follower networks," in *PROCEEDINGS OF THE 19TH INTERNATIONAL CONFERENCE ON INFORMATICS IN CONTROL, AUTOMATION AND ROBOTICS (ICINCO)*, G. Gini, H. Nijmeijer, W. Burgard, and D. Filev, Eds., 19th International Conference on Informatics in Control, Automation and Robotics (ICINCO), Lisbon, PORTUGAL, JUL 14-16, 2022, 2022, pp. 330–339, ISBN: 978-989-758-585-2. DOI: 10.5220/0011320200003271.

- ——, "Distributed formation control of drones with onboard perception," *IEEE-ASME TRANSACTIONS ON MECHATRONICS*, vol. 27, no. 5, pp. 3121–3131, Oct. 2022, ISSN: 1083-4435. DOI: 10.1109/TMECH.2021.3110660.

- L. Guo, Y. Guo, J. Wang, and T. Wei, "Ultraviolet communication technique and its application," *JOURNAL OF SEMICONDUCTORS*, vol. 42, no. 8, Aug. 2021, ISSN: 1674-4926. DOI: 10.1088/1674-4926/42/8/081801.

- Y. Zheng, Z. Chen, D. Lv, Z. Li, Z. Lan, and S. Zhao, "Air-to-air visual detection of micro-uavs: An experimental evaluation of deep learning," *IEEE ROBOTICS AND AUTOMATION LETTERS*, vol. 6, no. 2, pp. 1020–1027, Apr. 2021, ISSN: 2377-3766. DOI: 10.1109/LRA.2021.3056059.

- E. Balestrieri, P. Daponte, L. De Vito, and F. Lamonaca, "Sensors and measurements for unmanned systems: An overview," *SENSORS*, vol. 21, no. 4, Feb. 2021. DOI: 10.3390/s21041518.

- M. Shafiq and A. M. Khan, "Formation control of multiple uavs using pid control approach," *INTERNATIONAL JOURNAL OF MODELLING IDENTIFICATION AND CONTROL*, vol. 39, no. 4, pp. 340–349, 2021, ISSN: 1746-6172. DOI: 10.1504/IJMIC.2021.123802.

- M. Lin and K. Lee, "Outdoor target positioning using wii remote ir camera and signal modulation," *SENSORS*, vol. 20, no. 8, Apr. 2020. DOI: 10.3390/s20082163.

- H. Xu, L. Wang, Y. Zhang, K. Qiu, and S. Shen, "Decentralized visual-inertial-uwb fusion for relative state estimation of aerial swarm," in *2020 IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION (ICRA)*, ser. IEEE International Conference on Robotics and Automation ICRA, IEEE International Conference on Robotics and Automation (ICRA), ELECTR NETWORK, MAY 31-JUN 15, 2020, 2020, pp. 8776–8782, ISBN: 978-1-7281-7395-5. DOI: 10.1109/icra40945.2020.9196944.

- J. Sun, Y. Wu, X. Lu, and Y. Feng, "Cascade improved visual detection and distance estimation in formation of micro unmanned aerial vehicles," in *PROCEEDINGS OF THE 11TH INTERNATIONAL CONFERENCE ON MODELLING, IDENTIFICATION AND CONTROL (ICMIC2019)*, R. Wang, Z. Chen, W. Zhang, and Q. Zhu, Eds., ser. Lecture Notes in Electrical Engineering, 11th International Conference on Modelling, Identification and Control (ICMIC), Tianjin, PEOPLES R CHINA, JUL 13-15, 2019, vol. 582, 2020, pp. 755–765, ISBN: 978-981-15-0474-7; 978-981-15-0473-0. DOI: 10.1007/978-981-15-0474-7_71.

- R. Opromolla, G. Inchingolo, and G. Fasano, "Airborne visual detection and tracking of cooperative uavs exploiting deep learning," *SENSORS*, vol. 19, no. 19, Oct. 2019. DOI: 10.3390/s19194332.

- Y. Li, S. Zahran, Y. Zhuang, Z. Gao, Y. Luo, Z. He, L. Pei, R. Chen, and N. El-Sheimy, "Imu/magnetometer/barometer/mass-flow sensor integrated indoor quadrotor uav localization with robust velocity updates," *REMOTE SENSING*, vol. 11, no. 7, Apr. 2019. DOI: 10.3390/rs11070838.

- K. Maeda, Y. Funabora, S. Doki, and K. Doki, "Flight path planning of multiple uavs for robust localization near infrastructure facilities," in *IECON 2018 - 44TH ANNUAL CONFERENCE OF THE IEEE INDUSTRIAL ELECTRONICS SOCIETY*, ser. IEEE Industrial Electronics Society, 44th Annual Conference of the IEEE Industrial-Electronics-Society (IECON), Washington, DC, OCT 20-23, 2018, 2018, pp. 2522–2527, ISBN: 978-1-5090-6684-1.

> 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 *14th IEEE International Conference on Automation Science and Engineering (CASE 2018)*, © 2018 IEEE. Reprinted, with permission., 2018; Citations according to Web of Science, accessed 4.12.2023.

- X. Si, G. Xu, M. Ke, H. Zhang, K. Tong, and F. Qi, "Relative localization within a quadcopter unmanned aerial vehicle swarm based on airborne monocular vision," *DRONES*, vol. 7, no. 10, Oct. 2023. DOI: 10.3390/drones7100612.

- L. Ma, D. Meng, X. Huang, and S. Zhao, "Vision-based formation control for an outdoor uav swarm with hierarchical architecture," *IEEE ACCESS*, vol. 11, pp. 75 134–75 151, 2023, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2023.3296603.

- G. A. Amran, S. Wang, M. A. A. Al-Qaness, S. A. H. Mohsan, R. Abbas, E. Ghaleb, S. Al-shathri, and M. A. Elaziz, "Efficient and secure wifi signal booster via unmanned aerial vehicleswifi repeater based on intelligence based localization swarm and blockchain," *MICROMACHINES*, vol. 13, no. 11, Nov. 2022. DOI: 10.3390/mi13111924.

- M. Foughali and A. Zuepke, "Formal verification of real-time autonomous robots: An interdisciplinary approach," *FRONTIERS IN ROBOTICS AND AI*, vol. 9, Apr. 2022, ISSN: 2296-9144. DOI: 10.3389/frobt.2022.791757.

- Z. Wang, Y. Ng, J. Henderson, and R. Mahony, "Smart visual beacons with asynchronous optical communications using event cameras," in *2022 IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS AND SYSTEMS (IROS)*, ser. IEEE International Conference on Intelligent Robots and Systems, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, JAPAN, OCT 23-27, 2022, 2022, pp. 3793–3799, ISBN: 978-1-6654-7927-1. DOI: 10.1109/IROS47612.2022.9982016.

V. Walter, V. Spurný, M. Petrlík, T. Báča, D. Žaitlík, and M. Saska, "Extinguishing of Ground Fires by Fully Autonomous UAVs Motivated by the MBZIRC 2020 Competition," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, © 2021 IEEE. Reprinted, with permission., Jun. 2021, pp. 787–793. DOI: 10.1109/ICUAS51884.2021.9476723; Citations according to Scopus, accessed 4.12.2023.

- G. Roggi, S. Meraglia, and M. Lovera, "Leonardo drone contest autonomous drone competition: Overview, results, and lessons learned from politecnico di milano team," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 108, no. 2, 2023, cited By 0. DOI: 10.1007/s10846-023-01855-w
  [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85161828006&doi=10.1007%2fs10846-023-01855-w&partnerID=40&md5=95f007df4475d9afcdf34f677a43fcc5.

- H. Sun, H. Yan, M. Hassanalian, J. Zhang, and A. Abdelkefi, "Uav platforms for data acquisition and intervention practices in forestry: Towards more intelligent applications," *Aerospace*, vol. 10, no. 3, 2023, cited By 2. DOI: 10.3390/aerospace10030317
  [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85151498994&doi=10.3390%2faerospace10030317&partnerID=40&md5=d67563e1c510613a776b06e977c723df.

- J. Welgemoed, R. Phillips, and T. Niekerk, "An autonomous multirotor uav for security systems in rural south africa," cited By 0, 2023, pp. 12–18. DOI: 10.1109/ICMIMT59138.2023.10200559
  [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85168762774&doi=10.1109%2fICMIMT59138.2023.10200559&partnerID=40&md5=9cd292b7bf5edb3a1a94763d711c2148.

- L. Fagundes-Junior, C. Barcelos, D. Gandolfo, and A. Brandao, "Bdp-uaifly system: A platform for the robocup brazil open flying robot trial league," cited By 0, 2023, pp. 1021–1028. DOI: 10.1109/ICUAS57906.2023.10156168
  [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85165674245&doi=10.1109%2fICUAS57906.2023.10156168&partnerID=40&md5=45298cd2bac7beefe49a14a1d269883a.

- I. Alsammak, M. Mahmoud, H. Aris, M. Alkilabi, and M. Mahdi, "The use of swarms of unmanned aerial vehicles in mitigating area coverage challenges of forest-fire-extinguishing activities: A systematic literature review," *Forests*, vol. 13, no. 5, 2022, cited By 6. DOI: 10.3390/f13050811
  [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85131096464&doi=10.3390%2ff13050811&partnerID=40&md5=3d91d8cdd2a1e26eed4b809dcbd8de81.

- M. Khachumov and V. Khachumov, "Optimization models of uav route planning for forest fire monitoring," cited By 1, 2022, pp. 272–277. DOI: 10.1109/RusAutoCon54946.2022.9896260
  [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85140890489&doi=10.1109%2fRusAutoCon54946.2022.9896260&partnerID=40&md5=a9ed52a6dad1e137e325f0015a59622a.

- G. Roggi, S. Meraglia, and M. Lovera, "Leonardo drone contest 2021: Politecnico di milano team architecture," cited By 4, 2022, pp. 191–196. DOI: 10.1109/ICUAS54217.2022.9836103
  [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85136113592&doi=10.1109%2fICUAS54217.2022.9836103&partnerID=40&md5=07d52f906f890c62de31c8be33597d25.

J. Horyna, V. Walter, and M. Saska, "UVDAR-COM: UV-Based Relative Localization of UAVs with Integrated Optical Communication," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, Jun. 2022. DOI: `10.1109/ICUAS54217.2022.9836151`; Citations according to Web of Science, accessed 8.1.2024.

- M. Catellani and L. Sabattini, "Distributed control of a limited angular field-of-view multi-robot system in communication-denied scenarios: A probabilistic approach," *IEEE ROBOTICS AND AUTOMATION LETTERS*, vol. 9, no. 1, pp. 739–746, Jan. 2024, ISSN: 2377-3766. DOI: `10.1109/LRA.2023.3337694`.

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*, pp. 1–17, 2022. DOI: `https://doi.org/10.1007/s10514-022-10066-5`; Citations according to Web of Science, accessed 8.1.2024.

- M. Catellani and L. Sabattini, "Distributed control of a limited angular field-of-view multi-robot system in communication-denied scenarios: A probabilistic approach," *IEEE ROBOTICS AND AUTOMATION LETTERS*, vol. 9, no. 1, pp. 739–746, Jan. 2024, ISSN: 2377-3766. DOI: `10.1109/LRA.2023.3337694`.

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 *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, Jun. 2021, pp. 570–576. DOI: `10.1109/ICRA48506.2021.9561284`; Citations according to Web of Science, accessed 8.1.2024.

- 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, May 2024, ISSN: 0957-4174. DOI: `10.1016/j.eswa.2023.122694`.

- 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, Nov. 2023, ISSN: 1270-9638. DOI: `10.1016/j.ast.2023.108665`.

- 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, Oct. 2023. DOI: `10.3390/robotics12050139`.

- 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, Sep. 2023, ISSN: 2377-3766. DOI: `10.1109/LRA.2023.3296037`.

- 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, Nov. 2022. DOI: `10.3390/f13111889`.

- 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 *2022 IEEE/RSJ INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS AND SYSTEMS (IROS)*, ser. IEEE International Conference on Intelligent Robots and Systems, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, JAPAN, OCT 23-27, 2022, IEEE; Royal Soc Japan; IEEE Robot & Automat Soc; IES; SICE; New Technol Fdn, 2022, pp. 380–387, ISBN: 978-1-6654-7927-1. DOI: `10.1109/IROS47612.2022.9981115`.

F. Novák, V. Walter, P. Petráček, T. Báča, and M. Saska, "Fast collective evasion in self-localized swarms of unmanned aerial vehicles," *Bioinspiration & Biomimetics*, vol. 16, no. 6, p. 066 025, Nov. 2021. DOI: `10.1088/1748-3190/ac3060`
[Online]. Available: `https://doi.org/10.1088/1748-3190/ac3060`; Citations according to Web of Science, accessed 8.1.2024.

- L. Kong, L. Wang, Z. Cao, and X. Wang, "Resilience evaluation of uav swarm considering resource supplementation," *RELIABILITY ENGINEERING & SYSTEM SAFETY*, vol. 241, Jan. 2024, ISSN: 0951-8320. DOI: `10.1016/j.ress.2023.109673`.

Bibliography

[1] 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," *JIRS*, vol. 102, no. 26, pp. 1–28, 1 2021.

[2] J. Gu, T. Su, Q. Wang, X. Du, and M. Guizani, "Multiple moving targets surveillance based on a cooperative network for multi-uav," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 82–89, 2018. DOI: 10.1109/MCOM.2018.1700422.

[3] S. Gupte, P. I. T. Mohandas, and J. M. Conrad, "A survey of quadrotor unmanned aerial vehicles," in *2012 Proceedings of IEEE Southeastcon*, 2012, pp. 1–6. DOI: 10.1109/SECon.2012.6196930.

[4] M. Petrlík, T. Báča, D. Heřt, M. Vrba, T. Krajník, 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. DOI: 10.1109/LRA.2020.2970980.

[5] P. Petráček, V. Krátký, M. Petrlík, T. Báča, R. Kratochvíl, 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. DOI: 10.1109/LRA.2021.3098304.

[6] N. Zhao, W. Lu, M. Sheng, Y. Chen, J. Tang, F. R. Yu, and K.-K. Wong, "Uav-assisted emergency networks in disasters," *IEEE Wireless Communications*, vol. 26, no. 1, pp. 45–51, 2019. DOI: 10.1109/MWC.2018.1800160.

[7] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, Z. Dou, E. Almaita, I. Khalil, N. S. Othman, A. Khreishah, and M. Guizani, "Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges," *IEEE Access*, vol. 7, pp. 48 572–48 634, 2019. DOI: 10.1109/ACCESS.2019.2909530.

[8] A. Ollero, M. Tognon, A. Suarez, D. Lee, and A. Franchi, "Past, present, and future of aerial robotic manipulators," *IEEE Transactions on Robotics*, vol. 38, no. 1, pp. 626–645, 2022. DOI: 10.1109/TRO.2021.3084395.

[9] Q. Ali, N. Gageik, and S. Montenegro, "A review on distributed control of cooperating mini uavs," *International Journal of Artificial Intelligence & Applications*, vol. 5, pp. 1–13, 2014 [Online]. Available: https://api.semanticscholar.org/CorpusID:58709123.

[10] R. Li and Y. Li, "Localization of leader-follower formations using kinect and rtk-gps," in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, Dec. 2014, pp. 908–913. DOI: 10.1109/ROBIO.2014.7090448.

[11] D. A. Mercado, R. Castro, and R. Lozano, "Quadrotors flight formation control using a leader-follower approach," in *(ECC 2013)*, Jul. 2013, pp. 3858–3863. DOI: 10.23919/ECC.2013.6669637.

[12] T. Chen, Q. Gao, and M. Guo, "An improved multiple uavs cooperative flight algorithm based on leader follower strategy," in *(CCDSC)*, Jun. 2018, pp. 165–169. DOI: 10.1109/CCDC.2018.8407124.

[13] S. van der Helm, M. Coppola, K. N. McGuire, and G. C. H. E. de Croon, "On-board range-based relative localization for micro air vehicles in indoor leader–follower flight," *Autonomous Robots*, 2019, ISSN: 1573-7527. DOI: 10.1007/s10514-019-09843-6.

[14] M. Vrba, V. Walter, V. Pritzl, M. Pliska, T. Báča, V. Spurný, D. Heřt, and M. Saska, *On onboard lidar-based flying object detection*, 2023. arXiv: 2303.05404 [cs.RO].

[15] R. Tron, J. Thomas, G. Loianno, J. Polin, V. Kumar, and K. Daniilidis, "Vision-based formation control of aerial vehicles," in *Robotics: Science and Systems*, 2014.

[16] M. Saska, T. Báča, J. Thomas, J. Chudoba, L. Preucil, T. Krajník, 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.

[17] K. Boudjit and C. Larbes, "Detection and target tracking with a quadrotor using fuzzy logic," in *8th International Conference on Modelling, Identification and Control (ICMIC)*, 2016. DOI: `10.1109/ICMIC.2016.7804285`.

[18] V. Dhiman, J. Ryde, and J. J. Corso, "Mutual localization: Two camera relative 6-dof pose estimation from reciprocal fiducial observation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013. DOI: `10.1109/IROS.2013.6696524`.

[19] P. Giguere, I. Rekleitis, and M. Latulippe, "I see you, you see me: Cooperative localization through bearing-only mutually observing robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012.

[20] O. De Silva, G. K. I. Mann, and R. G. Gosine, "An ultrasonic and vision-based relative positioning sensor for multirobot localization," *IEEE Sensors Journal*, vol. 15, no. 3, pp. 1716–1726, 2015. DOI: `10.1109/JSEN.2014.2364684`.

[21] 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 *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013. DOI: `10.1109/IROS.2013.6696456`.

[22] J. Liu and Y. Jin, "A comprehensive survey of robust deep learning in computer vision," *Journal of Automation and Intelligence*, 2023.

[23] 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. DOI: `10.1109/LRA.2020.2972819`.

[24] P. Petráček, V. Walter, T. Báča, 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, Dec. 2020. DOI: `https://doi.org/10.1088/1748-3190/abc6b3`.

[25] F. Novák, V. Walter, P. Petráček, T. Báča, and M. Saska, "Fast collective evasion in self-localized swarms of unmanned aerial vehicles," *Bioinspiration & Biomimetics*, vol. 16, no. 6, p. 066 025, Nov. 2021. DOI: `10.1088/1748-3190/ac3060`
[Online]. Available: `https://doi.org/10.1088/1748-3190/ac3060`.

[26] 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 *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, Jun. 2021, pp. 570–576. DOI: `10.1109/ICRA48506.2021.9561284`.

[27] 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*, pp. 1–17, 2022. DOI: `https://doi.org/10.1007/s10514-022-10066-5`.

[28] 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, Jul. 2019, ISSN: 2377-3766. DOI: `10.1109/LRA.2019.2901683`.

[29] V. Walter, M. Vrba, D. B. Licea, and M. Saska, "Distributed formation-enforcing control for uavs robust to observation noise in relative pose measurements," 2023. arXiv: `2304.03057 [cs.RO]`.

[30] 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. DOI: `10.1109/ACCESS.2022.3158758`.

[31] P. Stegagno, M. Cognetti, A. Franchi, and G. Oriolo, "Mutual localization using anonymous bearing measurements," in *IROS*, 2011.

[32] A. Franchi, C. Masone, V. Grabe, M. Ryll, H. H. Bülthoff, and P. R. Giordano, "Modeling and control of UAV bearing formations with bilateral high-level steering," *IJRR*, vol. 31, no. 12, 2012.

[33] M. Krizek, J. Horyna, and M. Saska, "Swarming of unmanned aerial vehicles by sharing distributed observations of workspace," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2022, pp. 300–309. DOI: `10.1109/ICUAS54217.2022.9836073`.

[34] T. Amorim, T. Nascimento, P. Petracek, G. de Masi, E. Ferrante, and M. Saska, "Self-organized uav flocking based on proximal control," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2021, pp. 1374–1382. DOI: `10.1109/ICUAS51884.2021.9476847`.

[35] 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, p. 1 006 786, 2022.

[36] D. Albani, T. Manoni, M. Saska, and E. Ferrante, "Distributed three dimensional flocking of autonomous drones," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 6904–6911. DOI: `10.1109/ICRA46639.2022.9811633`.

[37] J. Horyna, V. Kratky, E. Ferrante, and M. Saska, "Decentralized multi-robot velocity estimation for uavs enhancing onboard camera-based velocity measurements," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 11 570–11 577. DOI: `10.1109/IROS47612.2022.9981894`.

[38] M. A. Merzi, V. Gazi, G. Fedele, L. D'Alfonso, and A. Bono, "Target capturing in an ellipsoidal region with a swarm of quadrotor uavs," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, 2022, pp. 972–981.

[39] K. Jindal, A. Wang, D. Thakur, A. Zhou, V. Spurny, V. Walter, G. Broughton, T. Krajnik, M. Saska, and G. Loianno, "Design and deployment of an autonomous unmanned ground vehicle for urban firefighting scenarios," *Field Robotics*, vol. 1, pp. 186–202, 2022, ISSN: 2771-3989. DOI: `10.55417/fr.2022015`.

[40] P. Stegagno, M. Cognetti, G. Oriolo, H. H. Bülthoff, and A. Franchi, "Ground and aerial mutual localization using anonymous relative-bearing measurements," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1133–1151, Oct. 2016, ISSN: 1552-3098. DOI: `10.1109/TRO.2016.2593454`.

[41] W. Hönig, J. A. Preiss, T. K. S. Kumar, G. S. Sukhatme, and N. Ayanian, "Trajectory planning for quadrotor swarms," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 856–869, 2018. DOI: `10.1109/TRO.2018.2853613`.

[42] F. Liao, R. Teo, J. L. Wang, X. Dong, F. Lin, and K. Peng, "Distributed formation and reconfiguration control of vtol uavs," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 1, pp. 270–277, 2017. DOI: `10.1109/TCST.2016.2547952`.

[43] X. Dong, B. Yu, Z. Shi, and Y. Zhong, "Time-varying formation control for unmanned aerial vehicles: Theories and applications," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 1, pp. 340–348, 2015. DOI: `10.1109/TCST.2014.2314460`.

[44] V. Trianni, *Evolutionary swarm robotics: evolving self-organising behaviours in groups of autonomous robots*. Springer, 2008, vol. 108.

[45] G. F. Young, L. Scardovi, A. Cavagna, I. Giardina, and N. E. Leonard, "Starling flock networks manage uncertainty in consensus at low cost," *PLoS computational biology*, vol. 9, no. 1, e1002894, 2013.

[46] J. Horyna, V. Walter, and M. Saska, "UVDAR-COM: UV-Based Relative Localization of UAVs with Integrated Optical Communication," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, Jun. 2022. DOI: `10.1109/ICUAS54217.2022.9836151`.

[47] Y. Zheng, Z. Chen, D. Lv, Z. Li, Z. Lan, and S. Zhao, "Air-to-air visual detection of micro-uavs: An experimental evaluation of deep learning," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1020–1027, 2021, cited By 29. DOI: `10.1109/LRA.2021.3056059`
[Online]. Available: `https://www.scopus.com/inward/record.uri?eid=2-s2.0-85100705092&doi=10.1109%2fLRA.2021.3056059&partnerID=40&md5=4074331072bcd0a1bd4bd3d913474c36`.

[48] E. Vroon, J. Rojer, and G. de Croon, "Motion-based mav detection in gps-denied environments," in *Proceedings of the 12th International Micro Air Vehicle Conference*, 2021.

[49] J. Cortés and M. Egerstedt, "Coordinated control of multi-robot systems: A survey," *SICE JCMSI*, vol. 10, no. 6, pp. 495–503, 2017.

[50] D. Zelazo, A. Franchi, and P. R. Giordano, "Rigidity theory in SE(2) for unscaled relative position estimation using only bearing measurements," in *ECC*, 2014, pp. 2703–2708. DOI: 10.1109/ECC.2014.6862558.

[51] K. Lu, S.-L. Dai, and X. Jin, "Fixed-time rigidity-based formation maneuvering for nonholonomic multirobot systems with prescribed performance," *Transactions on Cybernetics*, pp. 1–13, 2022. DOI: 10.1109/TCYB.2022.3226297.

[52] F. Schiano, A. Franchi, D. Zelazo, and P. R. Giordano, "A rigidity-based decentralized bearing formation controller for groups of quadrotor UAVs," in *IROS*, 2016, pp. 5099–5106. DOI: 10.1109/IROS.2016.7759748.