Autonomous Drones in GNSS-Denied Environments: Results from the Leonardo Drone Contest

Original
Autonomous Drones in GNSS-Denied Environments: Results from the Leonardo Drone Contest / Godio, Simone; Marino, Francesco; Minervini, Alessandro; Primatesta, Stefano; Chiaberge, Marcello; Guglieri, Giorgio. - ELETTRONICO. - (2023). (Intervento presentato al convegno 10th International Workshop on Metrology for AeroSpace (MetroAeroSpace) tenutosi a Milan, Italy nel 19-21 June 2023) [10.1109/MetroAeroSpace57412.2023.10190003].

Availability:
This version is available at: 11583/2980661 since: 2023-08-30T09:23:55Z

Publisher:
IEEE

Published
DOI:10.1109/MetroAeroSpace57412.2023.10190003

Autonomous Drones in GNSS-Denied Environments: Results from the Leonardo Drone Contest

Simone Godio
Dept. of Mechanical and Aerospace Engineering
Politecnico di Torino
Torino, Italy
simone.godio@polito.it

Francesco Marino
Dept. of Mechanical and Aerospace Engineering
Politecnico di Torino
Torino, Italy
francesco_marino@polito.it

Alessandro Minervini
Dept. of Mechanical and Aerospace Engineering
Politecnico di Torino
Torino, Italy
alessandro.minervini@polito.it

Stefano Primatesta
Dept. of Mechanical and Aerospace Engineering
Politecnico di Torino
Torino, Italy
stefano.primatesta@polito.it

Marcello Chiaberge
Department of Electronics and Telecommunications
Politecnico di Torino
Torino, Italy
marcello.chiaberge@polito.it

Giorgio Guglieri
Dept. of Mechanical and Aerospace Engineering
Politecnico di Torino
Torino, Italy
giorgio.guglieri@polito.it

***Abstract*—The Leonardo Drone Contest is an autonomous drone competition that aims at finding innovative solutions for drones operating in a Global Navigation Satellite System (GNSS) denied environment. At the end of a three years cycle of the competition, in this paper a review of the identified system and conclusions made by the DRAFT team from Politecnico di Torino is presented. The authors aim at introducing the final solutions to the challenge in terms of hardware components, algorithms and development process. The proposed approach has been widely tested and validated, and it ranked second in the competition. The well-consolidated procedure, resulting from many iterations in the development cycle, has contributed to further improvements during the three-year challenge and can be helpful for anyone who desires to approach the problem of autonomous drones employed in smart cities contexts.**

***Index Terms*—Autonomy, Drones, GNSS-denied, Technology, Robotics, SLAM, Perception**

I. INTRODUCTION

The improvements in autonomous drone navigation, associated with the key role played by these technologies in multiple scenarios, have led many experts to increase the research effort in the autonomous navigation field [1]. The aim of improving autonomous drone navigation has encouraged many companies to organize contests where researchers and experts in this field are called to propose novel solutions according to state-of-the-art technologies.

Starting in 2016, the International Conference of Intelligent Robots and Systems (IROS) launched an annual challenge where participants are called to develop an Unmanned Aerial Vehicle (UAV) able to navigate a sequence of gates whose position was unknown until the race began. One of the most relevant approaches used in the challenge has been documented in [2]. In 2019, the collaboration between Lockheed Martin and Drone Racing League resulted in the AlphaPilot challenge, an open innovation contest of a race between autonomous flying drones. Some of the solutions presented to the challenge resulted in widely known papers [3], [4]. In 2022 at the International Conference of Unmanned Aircraft Systems (ICUAS) a UAV competition was organized. The competition aimed at finding innovative solutions to perform autonomous navigation in enclosed and unknown areas [5]. The second edition of the competition will be held in June 2023.

Some other contests were only based on simulation environments, such as Game of Drones organized by the NeurIPS conference. Also, the Co4AIR Marathon – Drone Racing Competition is an initiative by Co4AIR – Computers, Cognition and Communication, in which student teams implement machine vision, path planning and control algorithms in a MATLAB/Simulink$^{©}$ environment [6]. A more complete survey of drone challenges with their main aspects has been provided in [7].

One of the main challenges with autonomous drones is implementing efficient autonomous drone navigation in Global Navigation Satellite System (GNSS) denied environment. While GNSS autonomous navigation has been widely discussed [8], GNSS-denied environments provide limitations that need to be overcome to exploit drones' potentialities fully. Many tasks that drones are called to perform cannot make use of GNSS signal because it is degraded (rescue operations) [9] or absent (indoor applications) [10]. Many solutions have been introduced to overcome the limitations related to GNSS-denied environments [11], [12]: among various approaches, the most promising ones make use of SLAM techniques [13], [14] or ultra-wide-band (UWB) technology [15].

The aim of improving autonomous navigation in GNSS-denied and partially known environments is also the main goal of the Leonardo Drone Contest, the challenge launched by Leonardo Spa to encourage six Italian universities to deal with autonomous drone navigation based on SLAM techniques [16]. In this paper, the scientific objectives, the results of the DRones Autonomous Flight Team (DRAFT) and the format of the Leonardo Drone Contest are described. An overview of the developed system for this competition is provided and the solution to the proposed challenge in terms of hardware integration and developed algorithms is discussed.

The rest of the paper is organized as follows. Section II describes the race format and details the drone developed for the contest. Section III gives an overview of the implemented system, while Section IV presents the adopted solution and preliminary results. A discussion about the outcome of our research into smart mobility is provided in Section V. Finally, our conclusions are drawn in Section VI.

II. LEONARDO DRONE CONTEST

A. Race Format

The Leonardo Drone Contest [16] is a competition between universities that has been held annually since 2020. It is organized by one of the world leaders in aerospace, Leonardo Spa. The main challenge is to give the drone an increasing level of autonomy over the years. The competition setting is urban-like in a scaled indoor environment (length = 20m, width = 10m, and height = 3m), as shown in Fig. 1. While in the first years (2020, and 2021) the competition objectives were still single and limited to 3D reconstruction of the environment and precision landings, in the year 2022 (the last competition held) the competition rose considerably in complexity. The final goal was to develop a drone and a ground station system capable of reconstructing the 3D environment flown over, tracking moving objects accurately for a minimum time of 10 seconds, performing precision landings, and taking pictures of particular markers (ArUco) hidden between buildings.

Compared to previous years, the presence of unknown obstacles in the race course was also introduced, which had to be handled by the drone with active obstacle avoidance logic [17], [18], [19], [20] and also reconstructed in the 3D map. On this occasion, the competition included an initial phase where the drone had to chase the moving ground object, which often followed random movements. Once the chase was successfully completed, the jury would announce a sequence of cells to search for markers to photograph or cells to land in. A score was associated with each of these tasks. The final ranking was then drawn up based on the tasks successfully completed, the time taken to complete them, and extra points awarded for the 3D reconstruction of objects not known beforehand.

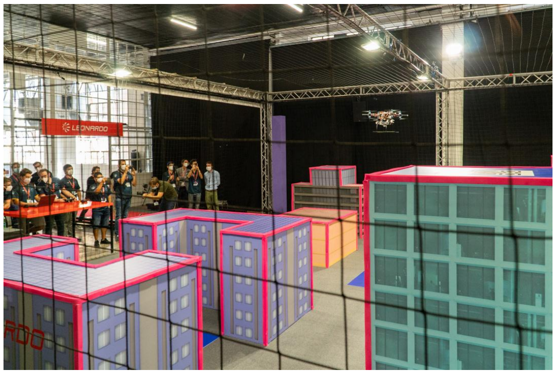


Fig. 1. Politecnico di Torino Team participating in Leonardo Drone Contest 2020.

B. Drone Specifications

The drone developed by the Politecnico di Torino to participate in the 2022 competition is shown in Fig. 2, and it has been named Agares by DRAFT team members. In particular, the developed quadcopter model was optimized in terms of weight and autonomy compared to previous years where a coaxial octa-copter was employed. The detailed characteristics are illustrated in Table I.

To give the aircraft the high autonomy capabilities required for competition, almost the entire payload shown in Table II is used to equip the drone with the sensors needed to perceive the external environment. The two Intel cameras are front mounted to respectively perform the tasks of extracting a point cloud in front of the drone and provide the on-board computer with a first rough estimate of the drone's position in space. This has the purpose of providing the aircraft with the necessary information on the object to be chased or the spots to land. Furthermore, the information coming from the single camera is also used in the logic of sensor fusion and pose estimation.

TABLE I
DRONE SPECIFICATIONS

Dimension	H = 0.35 m, W = S = 0.5 m
Weight [g]	2100
MTOW [g]	2600
Autonomy [min]	19
Payload [g]	500
Flight Controller [-]	Pixhawk 4 [21] - Ardupilot Firmware

C. Drone Model

This section presents an overview of the platform developed for the 2022 challenge, both from a hardware and software

Fig. 2. Agares flying during Leonardo Drone Contest 2022.

TABLE II
SENSORS

Stereo Camera	Intel D455 [22]
Tracking Camera	Intel T265 [23]
Mono-camera	M120 Raspberry Pi Camera [24]
Accel/Gyro (T265)	MPU6000
Accel/Gyro (Pixhawk 4)	ST Micro 16-bit gyroscope

point of view. The proposed solution results from three years of experience gained during previous challenges and related projects [25]. The software architecture instead is illustrated in Fig. 3. The same shows how it is divided into four main modules:

- Robot System Engineering (RSE): it contains the packages that deal with the interface with the flight controller (using the mavros node through the MAVlink protocol), the management files of the entire system, and the simulation in the Robot Operating System (ROS) and Gazebo environment.
- Deep Learning and Computer Vision (DLCV): it contains the packages that deal with analyzing the images coming from the cameras. The main packages contained within carry out the detection of markers (e.g. ArUco or cells of the competition area) through classical compute vision or neural networks developed ad hoc.
- Obstacle Avoidance and Motion Planning (OAMP): it contains packages that deal with global (Astar 3D ROS service) and local (Astar 3D dynamic) trajectory planning. Furthermore, inside there are the mission management scripts (task scheduler and task executor), the target tracking control logic, and the precision landing control logic.
- Simultaneous Localization And Mapping (SLAM): it contains the packages that deal with sensor fusion and therefore with pose estimation. As seen in Fig. 3, data from the mono-camera are also used. Furthermore, it manages point cloud data integrated with position data to generate a reliable reconstruction of the 3D environment and therefore the complete SLAM. For the mapping step an open-source ROS package, called RTAB-Map [26], was used, as shown below.

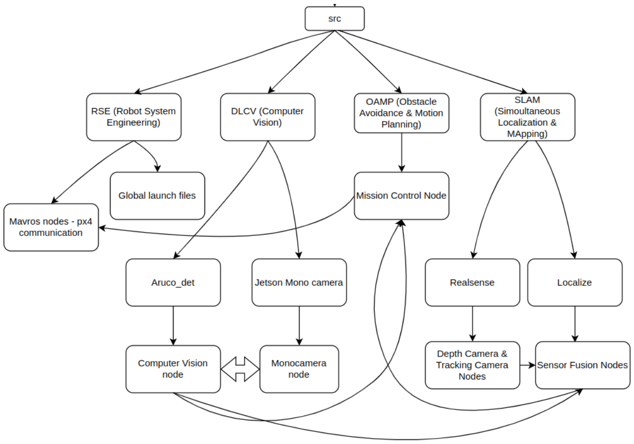


Fig. 3. Agares' software architecture.

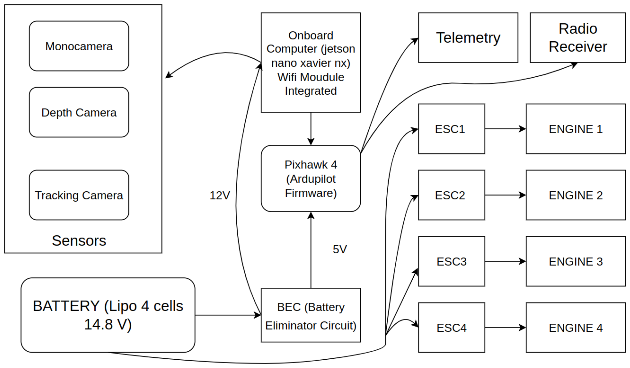


Fig. 4. Agares' hardware architecture.

In Fig. 4, the hardware architecture designed and built for the last competition is described. It can be seen that a Battery Eliminator Circuit (BEC) is needed in the electrical system to supply the correct voltage to the various consumers. In this case, the battery used is a 14.8 V / 4 cells setup which also powers the Electronic Speed Controllers (ESC); while the input voltage to the on-board computer (Jetson Xavier NX) can thus be fixed around 12 V. As shown, the main logic flow sees the data coming from the sensors as input to the on-board computer; once the data has been processed, the on-board computer provides the desired speed commands directly to the Pixhawk flight controller which manages the behavior of the engines and therefore the attitude and motion of the drone. Naturally, the radio receiver and telemetry used for pilot emergency control and data exchange with the flight controller are also present in the architecture.

III. SYSTEM OVERVIEW

A. Perception

The Perception of the drone is realized with two different camera sensors. The main navigation camera is the stereo camera (Intel® RealSense^{TM} D455) positioned in the direction of the drone's motion (forward-facing). This camera is used to estimate the distance of the object present in the trajectory of the drone itself. This camera is the main input of the obstacle avoidance algorithm that is able to estimate in real-time the amount of occluded space in the forward path of the platform and to suggest a collision resolution route to minimize the risk of impact during the navigation. Additionally, the stereo camera allows the detection of competition related targets on the obstacles. The second perception system is a mono-camera. This camera is mounted down-facing and it is implemented to allow precision landing and to support the Visual Inertial Odometry (VIO) algorithms by extracting some characteristic features of the ground to provide information about relative displacements. The two sensors work seamlessly to provide robustness in terms of multi-point-of-view capability. Additionally, much effort was directed into optimizing the algorithm using the raw images to operate computationally efficiently.

B. *State Estimation*

In an indoor application, the correct estimation of the robotic platform state is critical. The solution proposed in this paper takes advantage of a specific commercial sensor, i.e. the Intel® RealSense™ Tracking Camera T265. This sensor is used to gran-view cameras in a stereo set-up to extract features from the space at roughly 200 Hz frequency. Despite the sensor having been proven to be precise enough in navigation, experimental tests have identified a drift in the order of magnitude of two meters. This error is not acceptable for accomplishing contest tasks, so in order to compensate for the drifting Visual Inertial Odometry (VIO) estimate, two additional corrections have been implemented.

The first correction uses the presence of a specific feature in the competition ground. Since this feature was unique and known in position, a control loop has been applied to perform the correction of the estimated location of the drone wherever it is observed. This simple correction provides a discrete adjustment only in the proximity of the unique feature of interest.

The second correction is in principle similar to the first one but can be exploited during the entire mission. For the second correction we extract the ground pattern from the video stream of the mono-camera. Whenever a full set of four expected features is extracted, the algorithm tracks them over time to determine a relative drift. This information is then combined with the estimation from the T265 camera to calculate a final combined estimation of the two information.

Computing the state estimate with these two additional algorithms introduces latency in the order of 150 ms to the localization module, because of this it has been decided to decouple the localization from the control problem by feeding the full 200 Hz attitude estimation from the T265 to the controller. This solution allows a stable control of the drone and at the same time allows a reasonably quick correction of the localization required to perform accurate path planning.

C. *Tracking*

The possibility of tracking a specific movable object in the field is a competition requirement. This specific feature and software capability requires a reconfiguration of the purpose of some perception and control capabilities. To avoid redundant hardware, it has been decided to implement the tracking mode as part of the operations mode. During the tracking mode, the drone uses the down-facing mono camera to detect the movable object. The information is processed, and a relative drift from the centre of the drone is calculated. The controller is then able to incorporate the information of the error with respect to the target movable object and minimize it with a control scheme including of PID family. In particular, the chosen gains act on the target position and velocity. A tuning phase was performed both in a preliminary setting during tests and on the competition field to achieve the desired performance. Finally, the outputs of the control loops, i.e., a total thrust and angular velocity command, are sent to the rotors of the drone.

IV. Detailed Solution and Results

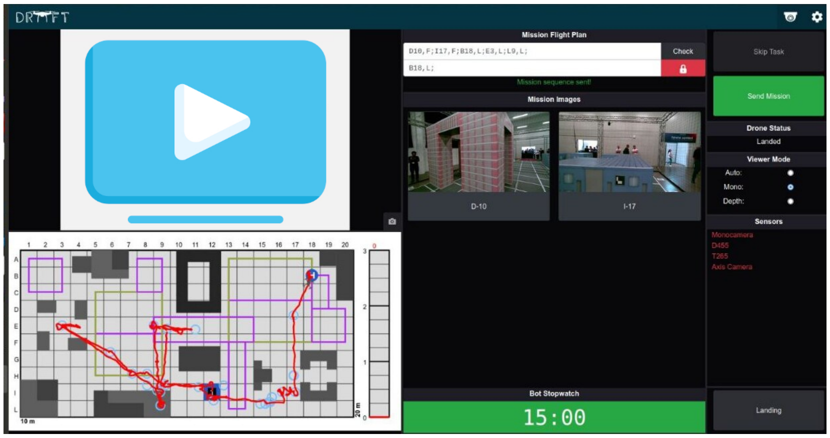
A. *Multi task planning*

The graphical user interface of the drone remote control station developed for the 2023 challenge is reported in Fig 5. The application was developed entirely on Javascript. A particular node was also developed using Python to interface the system with ROS, which handles the data flow between the flight and ground systems. As can be seen, the interface was designed to be simple and intuitive for the user. In the lower-left corner, the race map with the drone's estimated trajectory continuously updated during the flight immediately jumps out at the user. In the same map, it can be seen that the height of different obstacles has been associated with a more or less intense color intensity. In addition, the map also shows the possible search areas for the moving target, the takeoff pad, and the various waypoints followed by the drone during the mission. A column indicating the flight altitude immediately to the right of the map has also been included. In the upper left corner, however, the video stream that the user wishes to monitor can be selected and displayed: the front camera or the camera directed downward.

A special section called ”Mission Flight Plan” has been developed for sending the mission to the drone, as can be seen in the top right. Before sending the sequence to the drone, the ground control station also automatically checks the validity of user input to avoid communication errors. Immediately below the flight plan section can be seen the section where photos of the markers during the flight are saved; while still below is a timer to monitor the tracking time of the moving target. Finally, it is worth noting that in the lower-right corner there is also a section dedicated to monitoring the health of the various onboard sensors.

B. *Software In The Loop Development*

Before deployment on the real drone system, integration, and validation tests were carried out in a synthetic environ-



Mission Flight Plan
D10,F;I17,F;B18,L;E3,L;L9,L;
B18,L;
Mission sequence sent!

Drone Status
Landed

Viewer Mode
Auto: ○
Mono: ●
Depth: ○

Fig. 5. Ground Control Station platform developed from the team Draft members for Leonardo Drone Contest 2022.

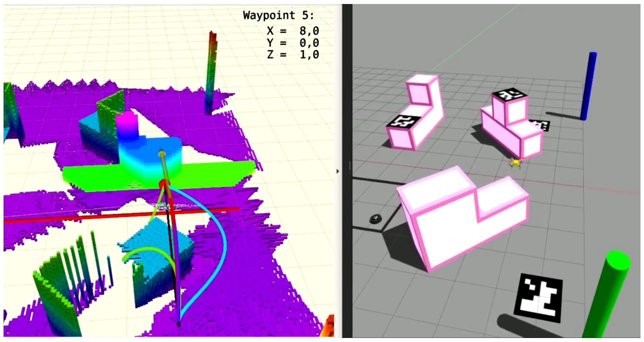
ment, as shown in Fig. 6. In this case, the environment was reconstructed with high fidelity using Gazebo, an open-source 3D robotics simulator.

First, the drone dynamics has been implemented in the Gazebo model through the provided Ardupilot plugin. Thanks to the continuous communication between the model and simulated dynamics, Gazebo is able to reproduce with high fidelity the model dynamics. To introduce the point cloud and evaluate the performance of the 3D mapping, Gazebo D435 plugin has been employed. This, although Gazebo is not a highly photo-realistic simulation system unlike Airsim [27], still allowed us to test and validate the main path planning, target tracking, precision landing, and obstacle avoidance packages with a reduced computational cost. Moreover, thanks to this approach, it was possible not only to validate individual nodes or packages, but the entire ROS-based software configuration shown in Fig. 3. Thus, in the real test phase, it was only necessary to refine certain parameters that differed between the virtual and simulated worlds, such as control logic gains, camera calibration and exposure parameters, and aircraft speed limits. Naturally, the integration path involved the deployment of a maximum of one or two code's features at a time on board the real aircraft to be validated during flight tests. This approach made it possible to achieve high efficiency and stability in the integration process. In particular, Fig. 6 shows the drone during simulation of the challenge where both the obstacle avoidance package of the local planner (left figure) and the 3D map reconstruction phase using RTAB-Map (right figure) were being validated.

SLAM algorithms were excluded from Software In The Loop (SITL) validation, due to the limitations of Gazebo in terms of photo-realistic simulation features. Nevertheless, SLAM algorithms were validated from recorded ROS bags and experimental testing.

V. APPLICATIONS TO SMART MOBILITY

According to the goals fixed by European Union Aviation Safety Agency (EASA) and Italian Civil Aviation Authority (ENAC) [28], advances in drones delivery (medical and goods) and Vertical Take-Off and Landing (eVTOL) aircraft for air mobility have been planned within 2030.



```
Waypoint 5:
 X =  8,0
 Y =  0,0
 Z =  1,0
```

Fig. 6. Software In The Loop Simulation Environment and Mapping.

The research project drawn in this paper is driven by this continental and national address, highlighting the crucial rule of a well-designed and consolidated development process to accomplish these demanding tasks. It is worth noting how the results reached in this paper are entirely tested and resulted from a three-year iteration on hardware set-up and algorithms. The development process proposed in this paper aimed at contributing to define a standardized process for autonomous drones flying in smart cities contexts.

The urban-like environment reproduced by the Leonardo Drone Contest proves the reliability of the proposed solution to deal with drawbacks related to visual navigation employed in smart cities contexts. The adopted methodology and tools demonstrate high effectiveness for the challenges proposed by the competition. Furthermore, the considerations made about the limitations of most widespread hardware employed for autonomous localization have been paired with the identified solutions. A similar approach to those proposed in this paper can be followed for VIO based navigation and it can be generalized to eVTOL aircraft employed for air mobility. Indeed, while new technologies and innovative algorithms have been introduced, a standardized development process is required to deal with the incoming air mobility challenges.

VI. CONCLUSIONS

In this paper, the second-place solution of the third edition of the Leonardo Drone Contest has been described. In particular, the aim of the contest, its rules and the main ideas behind the design process have been described. The authors presented the platform and provided an overview of the hardware and the software, with a particular focus on the equipped sensors and their architecture.

The identified sensor equipment provides a reliable solution to perform autonomous drone navigation in GNSS-denied environments. Although VIO still has weaknesses as concerns the state estimation problem, the loop closure suggested in this paper allows to reduce the localization error. The T265 camera limitations in terms of accuracy have been overcome thanks to the developed algorithms and the described corrections, and the resultant increase in the computation cost has been handled.

The SITL development was crucial to test the software architecture and algorithms, and the chosen tools and methodology resulted to be successful to accomplish the tasks proposed by the challenge.

According to these aims, future works will be focused on the introduction of a ground agent to accomplish more demanding requirements, especially in terms of 3D mapping reconstruction.

VII. ACKNOWLEDGEMENT

The authors would like to acknowledge the contribution from the students of the DRAFT[1] (a student team) of the Politecnico di Torino, for supporting the research activity. This work has been developed with the contribution of the Politecnico di Torino Interdepartmental Centre for Service Robotics PIC4SeR[2].

REFERENCES

[1] T. Elmokadem and A. V. Savkin, "Towards fully autonomous uavs: A survey," *Sensors*, vol. 21, no. 18, p. 6223, 2021.

[2] P. Geneva, K. Eckenhoff, W. Lee, Y. Yang, and G. Huang, "Openvins: A research platform for visual-inertial estimation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 4666–4672.

[3] P. Foehn, D. Brescianini, E. Kaufmann, T. Cieslewski, M. Gehrig, M. Muglikar, and D. Scaramuzza, "Dynamics modeling and analysis of a flexible-base space robot for capturing large flexible spacecraft," *Multibody System Dynamics*, vol. 32, no. 3, pp. 357–401, 2014.

[4] C. D. Wagter, F. Paredes-Vallé , N. Sheth, and G. de Croon, "The sensing, state-estimation, and control behind the winning entry to the 2019 artificial intelligence robotic racing competition," *Field Robotics*, vol. 2, no. 1, pp. 1263–1290, mar 2022.

[5] 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)*, 2022, pp. 621–626.

[6] M. Dragomir, V.-M. Maer, and L. Buşoniu, "The co4air marathon – a matlab simulated drone racing competition," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2022, pp. 1219–1226.

[7] P. Foehn, D. Brescianini, E. Kaufmann, T. Cieslewski, M. Gehrig, M. Muglikar, and D. Scaramuzza, "Alphapilot: autonomous drone racing," *Autonomous Robots*, vol. 46, 01 2022.

[8] B. H. Y. Alsalam, K. Morton, D. Campbell, and F. Gonzalez, "Autonomous uav with vision based on-board decision making for remote sensing and precision agriculture," in *2017 IEEE Aerospace Conference*, 2017, pp. 1–12.

[9] J. Q. Cui, S. K. Phang, K. Z. Y. Ang, F. Wang, X. Dong, Y. Ke, S. Lai, K. Li, X. Li, F. Lin, J. Lin, P. Liu, T. Pang, B. Wang, K. Wang, Z. Yang, and B. M. Chen, "Drones for cooperative search and rescue in post-disaster situation," in *2015 IEEE 7th International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, 2015, pp. 167–174.

[10] J. Tiemann and C. Wietfeld, "Scalable and precise multi-uav indoor navigation using tdoa-based uwb localization," in *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2017, pp. 1–7.

[11] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2568–2599, 2019.

[12] N. Gyagenda, J. Hatilima, H. Roth, and V. Zhmud, "A review of gnss-independent uav navigation techniques," *Robotics and Autonomous Systems*, vol. 152, p. 104069, 02 2022.

[13] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[14] P. Geneva, K. Eckenhoff, and G. P. Huang, "A linear-complexity ekf for visual-inertial navigation with loop closures," *2019 International Conference on Robotics and Automation (ICRA)*, pp. 3535–3541, 2019.

[15] E. Sutera, V. Mazzia, F. Salvetti, G. Fantin, and M. Chiaberge, "Indoor point-to-point navigation with deep reinforcement learning and ultra-wideband," in *13th International Conference on Agents and Artificial Intelligence – Volume 1*, 02 2021.

[16] Leonardo S.p.a., "Drone contest: an open innovation challenge by leonardo," https://www.leonardo.com/en/innovation-technology/open-innovation/drone-contest, 2023, accessed: 2023-02-06.

[17] B. Zhou, F. Gao, L. Wang, C. Liu, and S. Shen, "Robust and efficient quadrotor trajectory generation for fast autonomous flight," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3529–3536, 2019.

[18] B. Zhou, F. Gao, J. Pan, and S. Shen, "Robust real-time uav replanning using guided gradient-based optimization and topological paths," *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1208–1214, 2019.

[19] B. Zhou, J. Pan, F. Gao, and S. Shen, "Raptor: Robust and perception-aware trajectory replanning for quadrotor fast flight," *IEEE Trans. Robot*, vol. 37, no. 6, p. 1992–2009, 2021.

[20] J. Kim, "Fast path planning of autonomous vehicles in 3d environments," *Applied Sciences*, vol. 12, no. 8, 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/8/4014

[21] Pixhawk, "Pixhawk products," https://pixhawk.org/products/, 2023, accessed: 2023-04-26.

[22] Intel® RealSense™, "Depth camera d455," https://www.intelrealsense.com/depth-camera-d455/, 2023, accessed: 2023-04-26.

[23] ——, "Tracking camera t265," https://www.intelrealsense.com/tracking-camera-t265/, 2023, accessed: 2023-04-26.

[24] ElektorStore, "M120 raspberry pi camera," https://www.elektor.com/raspberry-pi-high-quality-camera-module-m12-mount, 2023, accessed: 2023-04-26.

[25] G. Roggi, S. Meraglia, and M. Lovera, "Leonardo drone contest 2021: Politecnico di milano team architecture," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2022, pp. 676–685.

[26] M. Labbé and F. Michaud, "Rtab-map as an open-source lidar and visual simultaneous localization and mapping library for large-scale and long-term online operation," *Journal of Field Robotics*, vol. 36, no. 2, pp. 416–446, 2019.

[27] Microsoft®, "Project airsim for aerial autonomy," https://www.microsoft.com/en-us/ai/autonomous-systems-project-airsim?activetab=pivot1%3aprimaryr3, 2023, accessed: 2023-02-06.

[28] ENAC, "AAM national strategic plan (2021-2030) for the development of advanced air mobility in italy," https://www.enac.gov.it/, 2023, accessed: 2023-02-06.

---

[1] https://www.draftpolito.it/

[2] https://pic4ser.polito.it