

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/362284786>

# RMF-Owl: A Collision-Tolerant Flying Robot for Autonomous Subterranean Exploration

Conference Paper · June 2022

DOI: 10.1109/ICUAS54217.2022.9836115

CITATIONS

20

READS

44

7 authors, including:



**Paolo De Petris**

Norwegian University of Science and Technology

18 PUBLICATIONS 236 CITATIONS

SEE PROFILE



**Huan Dinh Nguyen**

University of Nevada, Reno

32 PUBLICATIONS 616 CITATIONS

SEE PROFILE



**Mihir Dharmadhikari**

BITS Pilani, K K Birla Goa

18 PUBLICATIONS 321 CITATIONS

SEE PROFILE



**Mihir Kulkarni**

Norwegian University of Science and Technology

20 PUBLICATIONS 289 CITATIONS

SEE PROFILE

# RMF-Owl: A Collision-Tolerant Flying Robot for Autonomous Subterranean Exploration

Paolo De Petris<sup>1</sup>, Huan Nguyen<sup>1</sup>, Mihir Dharmadhikari<sup>2</sup>, Mihir Kulkarni<sup>2</sup>,  
Nikhil Khedekar<sup>1</sup>, Frank Mascarich<sup>2</sup>, and Kostas Alexis<sup>1</sup>

**Abstract**—This work presents the design, hardware realization, autonomous exploration and object detection capabilities of RMF-Owl, a new collision-tolerant aerial robot tailored for resilient autonomous subterranean exploration. The system is custom built for underground exploration with focus on collision tolerance, resilient autonomy with robust localization and mapping, alongside high-performance exploration path planning in confined, obstacle-filled and topologically complex underground environments. Moreover, RMF-Owl offers the ability to search, detect and locate objects of interest which can be particularly useful in search and rescue missions. A series of results from field experiments are presented in order to demonstrate the system’s ability to autonomously explore challenging unknown underground environments.

## I. INTRODUCTION

Research in aerial robotics has presented exciting progress towards fast and agile navigation combined with accurate localization, precise mapping, and collision avoidance. This has allowed the execution of autonomous complex missions such as exploration of unknown subterranean environments [1, 2], infrastructure inspection [3, 4], search and rescue [5] and more. However, finding the maneuver to avoid all possible obstacles in the environment, or identifying the way to fit and fly through extremely confined settings is particularly challenging and, sometimes, not always possible. The problem is further amplified when one considers the implicit uncertainty of the robot localization and mapping especially in **GNSS-denied sensor-degraded environments**, as well as control inaccuracies. The more we push the limits of the flight envelope and the scope of autonomous missions, the harder the problem becomes for the onboard localization, mapping and collision-free navigation processes. The above motivated the process for the design and autonomy functionalities of the presented RMF-Owl aerial robot, shown in Figure 1, which is tailored to subterranean exploration.

Some of the greatest hazards on Earth can be found in subterranean environments: tunnel collapses, limited access to food and oxygen, maze-like cave systems, and oppressing darkness all represent serious dangers to those who work or operate underground. Utilization of autonomous robotic



Fig. 1. RMF-Owl: The collision-tolerant flying system designed for resilient autonomous subterranean exploration.

technologies to undertake significant workload is thus essential. Focusing specifically on the task of autonomous exploration and mapping of such settings and motivated by the **DARPA Subterranean Challenge** [6] activities, the operational domains can be distinguished in a) Tunnel-like environments such as underground mines, b) Urban underground infrastructure such as the subway network in metropolitan cities, and c) Cave networks often involving highly irregular geological features and unpredictable topologies. Aiming to support the goal of autonomous exploration in diverse subterranean settings, we develop and present RMF-Owl, an aerial robot tailored to navigate safely within narrow geometries, quickly within vast spaces, and seamlessly across complex topologies. The system is capable of resilient onboard localization and mapping, as well as autonomous exploration path planning. To optimize for its highest prioritized task, that of exploring through extremely narrow and obstacle-filled subterranean passages, RMF-Owl employs a collision-tolerant design to maximize its resilience. In this work, we detail the airframe and mechatronic design, outline the onboard localization and mapping system as well as the pipeline to detect and localize objects of interest, summarize the onboard exploration planner and present selected experimental results from underground environments.

The remainder of this paper is structured as follows: Section II outlines relevant work in the domain of collision-tolerant flying robots. The new robot’s design is presented in Section III, followed by an overview of the realized autonomy stack in Section IV. Field experimental results are detailed in V. Finally, conclusions are drawn in Section VI.

This material is based upon work supported by a) the Defense Advanced Research Projects Agency (DARPA) under Agreement No. HR00111820045, and b) the Research Council of Norway project SENTIENT, grant number 321435. The presented content and ideas are solely those of the authors.

<sup>1</sup>The authors are with the Autonomous Robots Lab, NTNU, O.S. Bragstads Plass 2D, 7034, Trondheim, NO <sup>2</sup>The authors are with the University of Nevada, Reno, 1664 N. Virginia, 89557, Reno, NV, USA  
paolo.de.petris@ntnu.no

## II. RELATED WORK

A niche community of researchers has worked in the domain of collision-tolerant flying robots. An overview of the literature allows for the derivation of a taxonomy of designs with respect to the way collision-tolerance is implemented and how it is exploited. Considering rigid frame designs, a set of researchers utilize rolling cage frames to support safety post-collision [7–9]. Focusing on enabling a dual modality of navigation, the authors in [10] propose a caged robot with the cage designed to double as a wheel on the ground. Optimizing for reduced weight and mechanical complexity, rigid designs not involving a rolling cage are presented in [11–13] including previous work from the authors. Considering the potential benefits of a compliant design in best absorbing the impact effects, a set of relevant designs have been proposed and follow both biomimetic and more classical engineered concepts [14–16]. Focusing on particularly lightweight protective designs, the authors in [17–19] present novel origami shroud designs. Emphasizing both compliance and the ability to re-orient in case of a crash and fall on the ground, the works in [20,21] present tensegrity-based soft collision-tolerant flying robots. Departing from multirotor and broadly rotorcraft designs, the contributions in [22, 23] present collision-tolerant fixed-wing designs including bioinspired shape-morphing for higher protection when not flying. Considering flapping-wing designs and miniaturized systems, a set of relevant collision-tolerant systems are presented in [24,25]. Beyond contributions in collision-tolerant aerial robot designs, a set of works focus on specialized autonomy functionalities exploiting impact resilience. The efforts in [26,27] propose path planning methods that are aware and exploit the resilience introduced by collision-tolerant frames. Considering the potential of using collisions as means of sensing, a set of relevant contributions are presented in [28–30] targeting to acquire both odometry and map information. Emphasizing multi-robot autonomy, the authors in [31] enable robust swarming without explicit collision avoidance mechanisms. Within the spectrum of the presented taxonomy of collision resilient aerial robots, this work presents a rigid non-rotating design with a particularly lightweight design that focuses on advanced perception and autonomy features tailored to subterranean exploration.

## III. SYSTEM DESIGN

This section overviews the design concept of RMF-Owl. RMF-Owl is part of the Resilient Micro Flyer (RMF) aerial robot family with a first more lightweight system presented in [11] and a version tailored to nuclear radiation characterization detailed in [32].

### A. Airframe

The frame of RMF-Owl is designed for prolonged endurance, agile flight, light weight and collision-tolerance. The main rigid component is fabricated using carbon-foam sandwich material (total width: 10mm, with 0.75mm carbon on each side). The carbon and foam core densities are approximately  $0.0011\text{g/mm}^3$  and  $0.00008758\text{g/mm}^3$

respectively, leading to a total airframe weight of 145g. The decision of leaving the front and the back open is to avoid obstruction with respect to the field of view of the front camera and respect symmetry. The platform integrates four T-Motor F60PRO IV V2.0 KV1950 DC Brushless motors controlled through their electronic speed controllers. Moreover, RMF-Owl integrates a PixRacer R15 as its main low-level autopilot unit offering attitude and thrust control. High-level position control and navigation autonomy is facilitated through a different processing board as detailed further below. The resulting dimensions of RMF-Owl, are  $38 \times 38 \times 24\text{cm}$  ( $L \times W \times H$ ) and the total weight, including all the sensing and processing components, cabling and its battery is 1.45kg. In this configuration the power to weight ratio is equal to 2 and the resulting flight time is 10 minutes.

### B. Electronics

As depicted in Figure 2, RMF-Owl is powered via a single Spektrum 4s 5000 mAh LiPo battery. When fully charged, this provides 16.8 V directly to the Power Distribution Board (PDB), a T-Motor F55A Pro II 4-in-1 ESC. Two separate lines are derived from the battery circuitry. The first powers an Ouster OS0 LiDAR sensor, with the usage of a Tracopower 30W Isolated DC-DC Converter (24V). The second, with the addition of a 5V/3A DC-DC, provides power to the onboard computer, described in detail below. The PDB delivers the proper power to all of the motors of RMF-Owl and to the autopilot, via a separate 5V built-in line. To enable visual detection in low-light/dark underground environments, the battery also powers a Lumenier sUAS High Lumen Overt LED, triggered via a PWM line directly from the autopilot to reduce power consumption and thus increase efficiency.

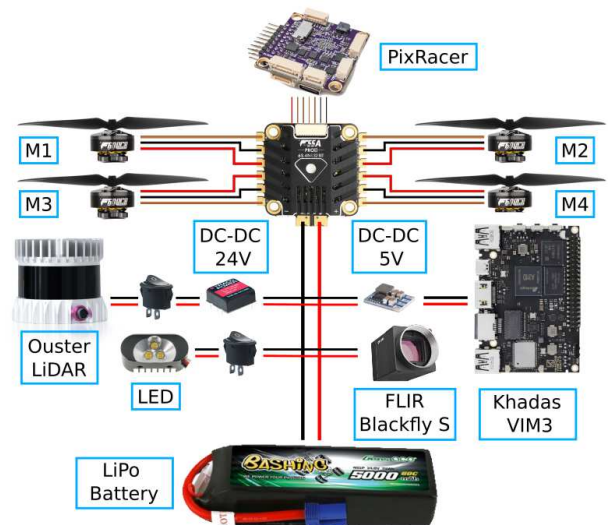


Fig. 2. Overview of the RMF-Owl electronics.

### C. Frame analysis under collision

In order to evaluate the mechanical properties of the designed carbon-foam frame, two studies are conducted. Firstly, a CAD simulation is conducted in order to identify possible breaking points on the structure and offer input for

possible future designs. Guided jointly by intuition and by the results of this first analysis, we then utilized duplicated parts of the current design, as well as a previous version of the frame - albeit thinner and thus a bit weaker - to get an understanding of what is the order of magnitude of forces that the frame can sustain and verify where the breaking points are to be expected.

1) *CAD Simulation*: In this test we applied different static forces to the model in simulation in an iterative fashion, looking for breaking points and possible regions that can be improved in next design iterations. As can be seen in Figure 5, the selected applied pressure points are top, bottom, lateral, frontal straight and frontal 45°. The Figure allows to derive an understanding as to which locations of the frame experience the maximum stress for certain directions of collision forces. The selection of collision points reflects the types of impact most commonly encountered during flight.

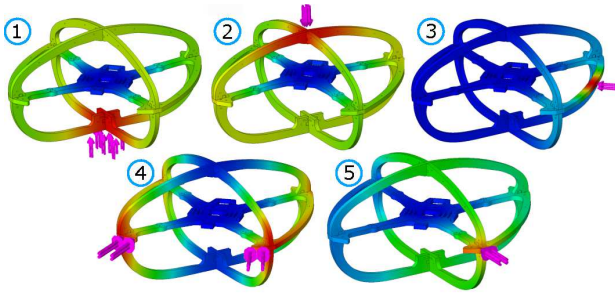


Fig. 3. CAD stress test-based qualitative analysis conducted on the frame of RMF-Owl. The static forces, represented by the pink arrows, are iteratively applied on the bottom (1), top (2), side (3), front (4) and front with an angle of 45° (5). The resulting frame displacement (reflecting stress) ranges from a minimum (blue) to a maximum value (red). Even in cases where such CAD simulations can present numerical errors in their results, the locations where maximum stress is applied and thus breaking is to be anticipated are expected to be qualitatively correct.

2) *Hydraulic Press Test*: In this second test the goal was to find the maximum force that the frame can sustain before breaking. Accordingly, a set of tests were conducted by placing: a duplicate part of the bottom of the frame (i), a duplicate part of the frame lateral protection (ii), and a previous - thinner - version of the frame in a normal operating position (iii) and also rolled it by 90° (iv) in between a piston of an industrial hydraulic press and a scale. An indicative setup example is depicted in Figure 4. Even though inaccuracies are introduced by the use of a frame with thinner carbon-foam arms and surrounding shroud and the utilized scale to estimate the force can present noise, this test allows to a) identify the approximate magnitude of forces that the frame can sustain, and b) verify the locations of maximum stress as also derived by the simulation studies. The maximum forces recorded during each of these experiments - as noted right before the frame broke - are reported in Table I.

#### D. High-Level Sensing and Processing Payload

The sensing payload of RMF-Owl is tailored to the goal of resilient autonomy in subterranean environments, while maintaining a very lightweight configuration. The system integrates a sensing suite consisting of an Ouster OS0-64



Fig. 4. Hydraulic press test experimental setup for case (iii) and (iv).

(i)	(ii)	(iii)	(iv)
285N	520N	285N	128N

TABLE I

MAXIMUM FORCE RECORDED BEFORE BREAKING IN EACH OF THE EXPERIMENTS WITH THE HYDRAULIC PRESS.

LiDAR sensor with Field Of View (FOV)  $FOV = [360, 90]^\circ$  and a Flir Blackfly S color camera with  $FOV = [85, 64]^\circ$ , interfaced with a Khadas VIM3 Pro Single Board Computer (SBC) incorporating  $\times 4$  2.2Ghz Cortex-A73 cores, paired with  $\times 2$  1.8Ghz Cortex-A53 cores implementing an A311D big-little architecture, alongside a Neural Processing Unit (NPU) offering 5.0 TOPS for dedicated neural network inference. The Khadas SBC interfaces the PixRacer autopilot to which it provides navigation commands. The SBC is also responsible for running all the localization and mapping, 3D occupancy mapping, path planning, as well as object detection and localization algorithms towards autonomous exploration in GPS-denied and confined environments.

## IV. RESILIENT AUTONOMOUS EXPLORATION

RMF-Owl implements a comprehensive autonomy stack that facilitates autonomous exploration and object search capabilities in complex subterranean environments.

### A. Architecture

RMF-Owl uses a high-level exploration path planner, described in Section IV-C, to plan feasible paths for the robot. The path is then tracked by a Proportional Integral Derivative (PID) controller described in Section IV-D and the low-level attitude and thrust commands are executed by the onboard autopilot. The odometry feedback of the robot and the map of its environment are provided by the localization and mapping solution presented in Section IV-B.

### B. Perception

The operational profile of RMF-Owl requires robust localization and mapping in subterranean environments. To this end we utilize the LiDAR odometry and mapping component of CompSLAM [33] which relies on the core ideas in [34]. The method utilizes 10Hz point cloud readings from the



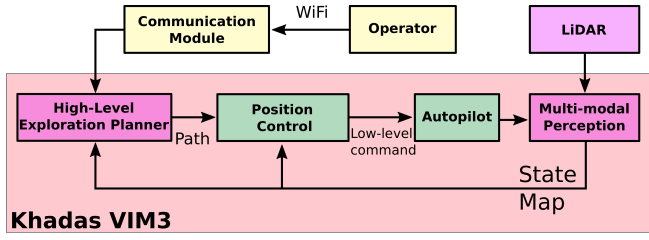


Fig. 5. Schematic overview of the core functional modules of RMF-Owl. The components inside the red shaded area are executed onboard the robot. Green color depicts robot-specific functionalities

onboard OS0-64 LiDAR. At full rate, it performs a scan-to-scan matching step, while at half rate (5Hz) it performs scan-to-submap matching. Accordingly, the robot estimates its pose  $\hat{\mathbf{p}}_k$  in the environment and simultaneously reconstructs a pointcloud model of the map  $\hat{\mathbf{m}}_k$  of the environment. The map is maintained in a  $1010 \times 1010 \times 510\text{m}$  cuboid around the initial position of the robot. If the robot gets too close to a face of the cuboid map, the entire map is shifted in a direction normal to the face such that the robot is always well contained within the map. To deal with computational limitations, the submap used for the scan-to-submap step has a maximum size of  $110 \times 110 \times 110\text{m}$ . For the scan-to-scan alignment step, 768 sharp corners and 1536 flat surface features are considered for each pointcloud observation. To handle very large environments, the software has the option to utilize only 32 of the 64 channels of the OS0-64 sensor thus reducing the computational cost. The reliable 5Hz pose update  $\hat{\mathbf{p}}_k$  is then fused with the onboard IMU using an Extended Kalman Filter based on the Multi-Sensor Fusion (MSF) framework [35]. The latter allows to acquire a smooth high frequency (100Hz) estimate  $\hat{\mathbf{p}}'_k$  which is then employed by the robot's position controller described in Section IV-D.

Beyond the requirement for robust localization and mapping, RMF-Owl also builds a volumetric map that is used to facilitate autonomous path planning, as well as object localization. By associating the incoming point cloud observations  $\mathbf{o}_k$  with the related pose estimates  $\hat{\mathbf{p}}_k$ , the system incrementally builds an occupancy map  $\hat{\mu}_k$  using Voxblox [36], a computationally efficient incremental mapping framework. For most missions, a voxel edge size of 0.2m is utilized.

### C. Exploration Path Planning

To achieve autonomous exploration, RMF-Owl utilizes a graph-based path planner called GBPlanner2 [37]. The method builds upon our previous work [1] and uses a bifurcated local/global planning architecture. The local planner is responsible for identifying efficient exploration paths that respect the robot motion and perception constraints. On the other hand, the global planner provides functionality for a) re-positioning the robot towards a previously perceived frontier of the exploration space when the local planner reports inability to find a path of significant exploration gain, and b) ensuring that the robot returns home within its endurance limits. Figure 6 provides an overview of the functionalities of GBPlanner2.

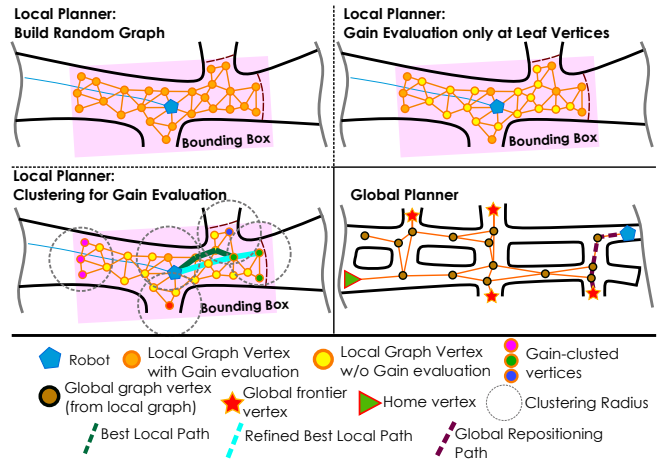


Fig. 6. Overview of GBPlanner2 functionalities.

In further detail, the local exploration planner of GBPlanner2 utilizes the Voxblox map  $\hat{\mu}_k$  to first sample vertices in free space within a local bounding box that is adaptively calculated using the aggregated point cloud around the current robot location. These vertices are connected by collision-free edges to build an undirected graph. Subsequently, the Dijkstra's algorithm is used to calculate the shortest paths starting from the current robot location. For each vertex  $\nu_i$  in the graph, an information gain, called **VolumetricGain**( $\nu_i$ ) =  $\epsilon_G \hat{\mu}_{k,unk}^{\nu_i}$  is calculated, where  $\hat{\mu}_{k,unk}^{\nu_i} \subset \hat{\mu}_k$  is the unknown part of the occupancy map that would lie in the robot's modeled sensor frustum if the robot was at the vertex  $\nu_i$  and  $\epsilon_G > 0$  is a tunable weight. This gain is then accumulated over a path to derive the total gain, called **ExplorationGain**, of that path. Since this is among the most computationally demanding processes - as it involves extensive raycasting - the planner provides the functionality to calculate the gain only at the leaf vertices of the shortest paths. Furthermore, the leaf vertices can be clustered using a radius  $\rho > 0$ , allowing to approximate the gain of some vertices based on the calculated gain of a nearby vertex. These approximations are necessary for computationally constrained Micro Aerial Vehicles like RMF-Owl to operate in large environments. Finally, the path having the highest **ExplorationGain** is selected, improved for safety as described in [1], and commanded as a reference to the position controller.

GBPlanner2 models the robot as a cuboid. A vertex is said to lie in free space if considering the robot's cuboid on it then all its voxels are free. However, this leads to sub-optimal behavior in narrow environments due to the map resolution. When the cuboid size is not a multiple of the map resolution along any axis, the actual volume checked for collision is larger than the size of the cuboid. Due to computational constraints, the map resolution used for RMF-Owl is relatively low (0.2 – 0.25m in most missions) and comparable to the size of the robot, as detailed in Section III. Hence, the above issue is prominent. To reliably traverse very narrow passages we exploit the system's collision tolerance to optimistically set the size of the modelled cuboid without any additional safety margins.

The global layer of GBPlanner2 maintains a sparse global graph built by incrementally appending only the high exploration gain paths from each local planning step, as well as the robot's state. The vertices in this graph that have a volumetric gain higher than a set threshold are characterized as "frontier vertices". When the local layer is unable to find a path with an **ExplorationGain** higher than a threshold for  $k$  consecutive iterations, the global planner is triggered to re-position the robot to one of the frontier vertices. The frontier vertex to re-position to is selected based on the frontier's volumetric gain and the exploration time remaining after reaching that frontier [1]. Furthermore, in each local planning iteration, the global planner finds the shortest path to the home location and commands it to the robot if the remaining endurance is only sufficient for it to return home.

#### D. Control

The position controller of RMF-Owl is a fixed-gain PID scheme, while the **yaw** controller utilizes a proportional scheme. This controller interfaces the low-level autopilot of the system which provides **roll** and **pitch** reference tracking, yaw rate reference tracking and thrust control. Let  $\mathcal{I}$  be the inertial frame, and  $\mathcal{V}$  the yaw-rotated inertial frame. The outputs of the position and yaw controllers are the commanded acceleration vectors expressed in  $\mathcal{I}$ ,  $[\mathcal{I}a_r^x, \mathcal{I}a_r^y, \mathcal{I}a_r^z]$ , and yaw rate  $\dot{\psi}_r$  which are then converted to the attitude-thrust command, as per [38], and forwarded to the low-level controller inside the autopilot running the PX4 stack:

$$\begin{aligned}\mathcal{I}a_r^x &= K_P^x(x_r - x) + K_I^x \int (x_r - x) dt \Big|_{t_{\min}^x}^{t_{\max}^x} + K_D^x(\dot{x}_r - \dot{x}) \quad (1) \\ \mathcal{I}a_r^y &= K_P^y(y_r - y) + K_I^y \int (y_r - y) dt \Big|_{t_{\min}^y}^{t_{\max}^y} + K_D^y(\dot{y}_r - \dot{y}) \\ \mathcal{I}a_r^z &= K_P^z(z_r - z) + K_I^z \int (z_r - z) dt \Big|_{t_{\min}^z}^{t_{\max}^z} + K_D^z(\dot{z}_r - \dot{z}) \\ \dot{\psi}_r &= K_P^\psi(\psi_r - \psi)\end{aligned}$$

where  $[x_r, y_r, z_r, \psi_r]$ ,  $[x, y, z, \psi]$  are the reference and estimated position and yaw angle of the robot, respectively expressed in  $\mathcal{I}$ ,  $I_{\min}^j, I_{\max}^j$ ,  $j \rightarrow x, y, z$  are the saturation minimum and maximum values of the control integrals, and  $K_P^j, K_I^j, K_D^j$ ,  $j \rightarrow x, y, z, \psi$  are the control gains.

#### E. Object Detection and Localization

RMF-Owl delivers functionality not only to volumetrically explore and map unknown subterranean environments but also to search, detect and localize objects of interest within them. The key elements of this procedure are outlined below.

1) *Visual Detection*: Object detection is achieved primarily on visual data using a YOLOv3 [39] model trained on a curated dataset of the objects of interest (called "artifacts"), collected using sensors from aerial and ground platforms using different cameras. Datasets were collected in buildings, abandoned tunnels, mines, natural caves with diverse light conditions and in the presence of obscurants. The object detector was trained using a combined dataset of 40,007 labels ranging across 8 classes of objects, namely i) a human survivor, ii) a fire extinguisher, iii) a drill, iv) a backpack, v) a vent, vi) a helmet, vii) a rope, and viii) a cellphone. The

trained model is computationally too expensive to be run on the CPU of the Khadas VIM3, however, its integrated 5.0 TOPS NPU provides dedicated compute for neural network inference. The trained weights file is converted to an NPU compatible format using the provided SDK. The weights of the fully trained neural network were converted from 32-bit floating point numbers to 8-bit integers to run on the NPU, which could then process camera images at 3Hz. A ROS interface was created to transfer the images from the camera to the NPU, and get detected artifact classes and corresponding bounding boxes as a result. This detection was then used by a multi-view consensus filter to estimate the location of the object.

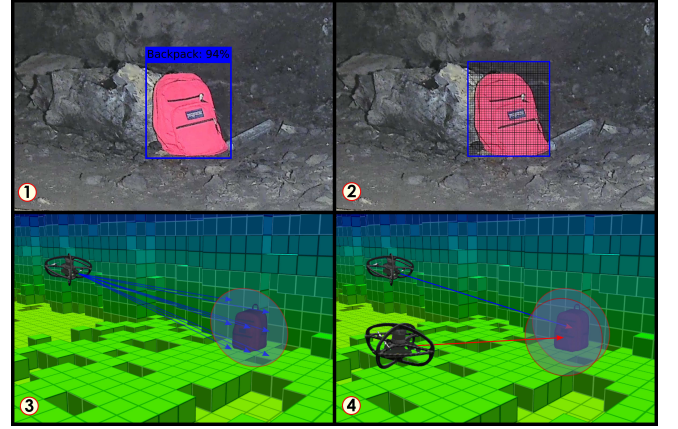


Fig. 7. Overview of the image detection and localization framework. The artifact is detected onboard (1), the bounding box is divided into pixels (2) and rays are cast into the map (3) to estimate the location of the object. Multiple detections are used over time to robustify the object detection, classification and localization (4).

2) *Localization*: Bounding boxes detected around the artifact are divided into a grid of pixels as shown in Figure 7. For each of these pixels, rays are cast into the robot's volumetric map  $\hat{\mu}_k$  using the camera intrinsic model and the camera-to-LiDAR extrinsic calibration. This results in a set of points that include the object and an area around it. The median point is selected as the estimated position  $a_j$  of the object, and a sphere with radius  $R_a$  is spawned around it. As the robot moves through the environment, subsequent detections are used to update  $a_j$  as an average of all detections projected into the sphere. For each class of object, separate binary Bayesian filters are used to estimate the probability of an object of a certain class being present in the sphere as detailed in [40]. Once the probability of a class exceeds a predefined threshold (specific to each class), the process is frozen for the corresponding sphere and the object is reported to the ground station.

3) *Detection of Bluetooth Devices*: A further capability of RMF-Owl is to detect devices that have their bluetooth modules on scan mode. Either for devices of known names or in discoverable mode, RMF-Owl allows to detect devices such as a cell phone or a computer, while for the location of the device the average robot's location during periods of detecting a certain device is considered.



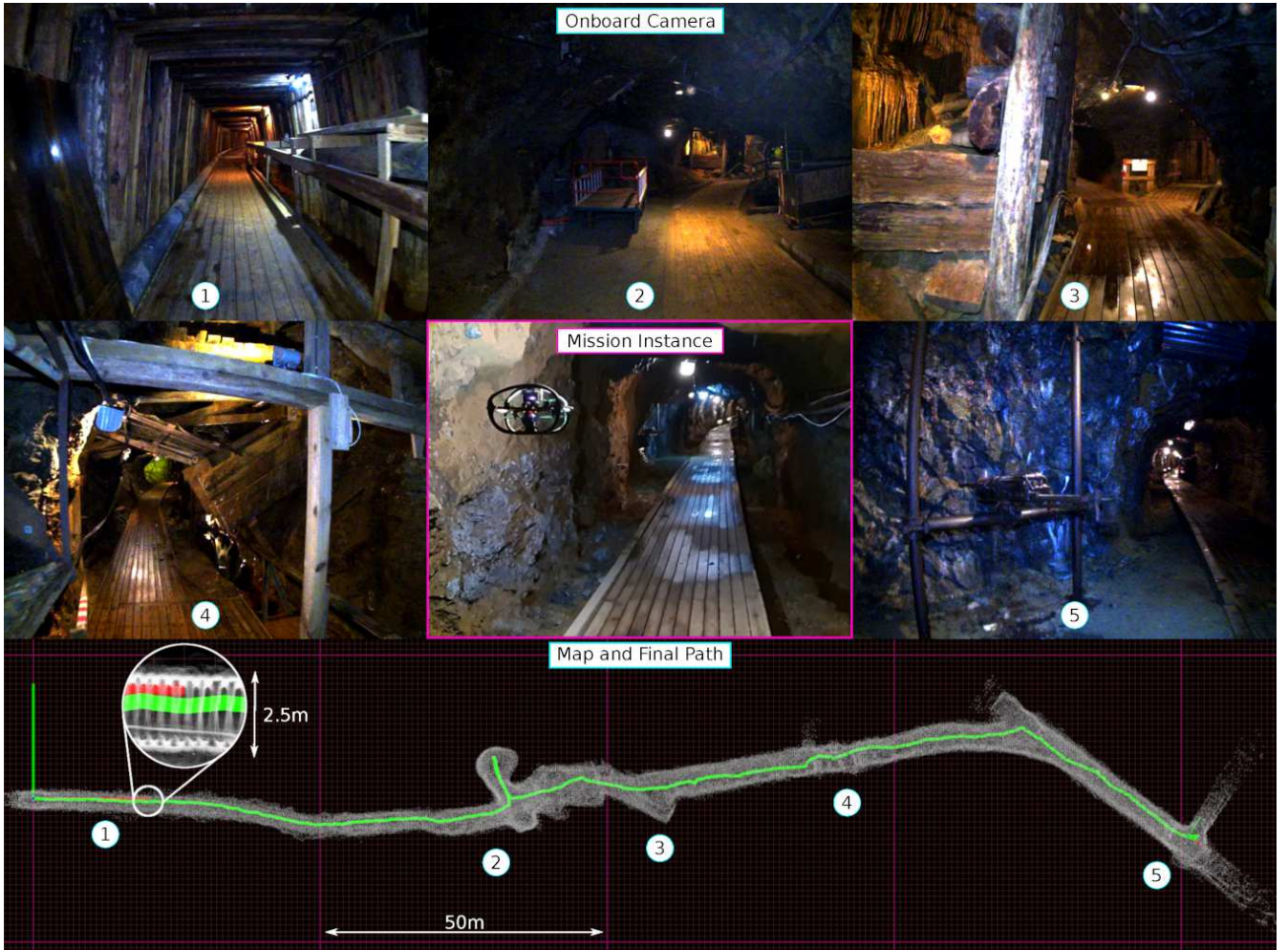


Fig. 8. Instances of an autonomous exploration mission inside the Løkken Mine. (1)-(5) instances from the onboard camera. Central image: RMF-Owl in flight. Bottom row: generated map, final executed path and corresponding location of the upper images. Starting from the main entrance, the robot is able to autonomously takeoff and smoothly plan collision-free paths inside the narrow corridor (1) and reach the first intersection (2). After a quick inspection of a room on the side of the main tunnel, RMF-Owl proceeds its course avoiding other crowded and wet sections (3), (4). Finally, the robot is commanded to safely land in a dry area instead of risking to land inside the drainage channel that is present in the next section of the tunnel (5).

4) *Reporting*: The ground station receives a report containing the class and location of a detected artifact, along with a summarized report of the class probabilities of the artifact. A downsampled image of the detected object is also sent to the ground station.

#### F. Communications and Networking

During mission deployments, RMF-Owl can communicate to send data and receive commands through 5GHz WiFi. Naturally, in the framework of subterranean exploration communications networking is very hard and the connection to the ground station is expected to be mostly not available. This is also one of the reasons for the emphasis of this work on resilient autonomy. However, when RMF-Owl operates in combination with other systems - as for example in the framework of deployments of Team CERBERUS in the DARPA Subterranean Challenge - it is possible to connect to a mesh of WiFi nodes if those are installed, deployed or ferried by other robots as detailed in [40]. When such a connection is available, the system shares mapping data, odometry status, battery level, object detection and localization reports, as well as other information.

## V. FIELD EXPERIMENTS

This section focuses on selected results from the deployment of RMF-Owl in different and challenging environments.

### A. Løkken Mine

The first field test took place in the Løkken Mine, located in the municipality of Orkland in Trøndelag, Norway. This environment was chosen specifically to test the resilience of RMF-Owl in narrow settings. As can be seen in Figure 8, the first section is less than 2.5m wide. The robot has no prior knowledge of the environment and it is commanded to explore as much as possible. From the mine entrance, the robot is able to autonomously takeoff (1), smoothly plan collision-free paths inside the narrow corridor (2) and reach the first intersection (3). After a quick inspection of a room on the side of the main tunnel, RMF-Owl proceeds its course avoiding other wet and crowded sections (4), (5). Finally, the robot is commanded to safely land in a dry area instead of risking landing inside the drainage channel present in the next section of the tunnel. The total travelled distance is more than 200m and the total flight time is 6.6min. Due



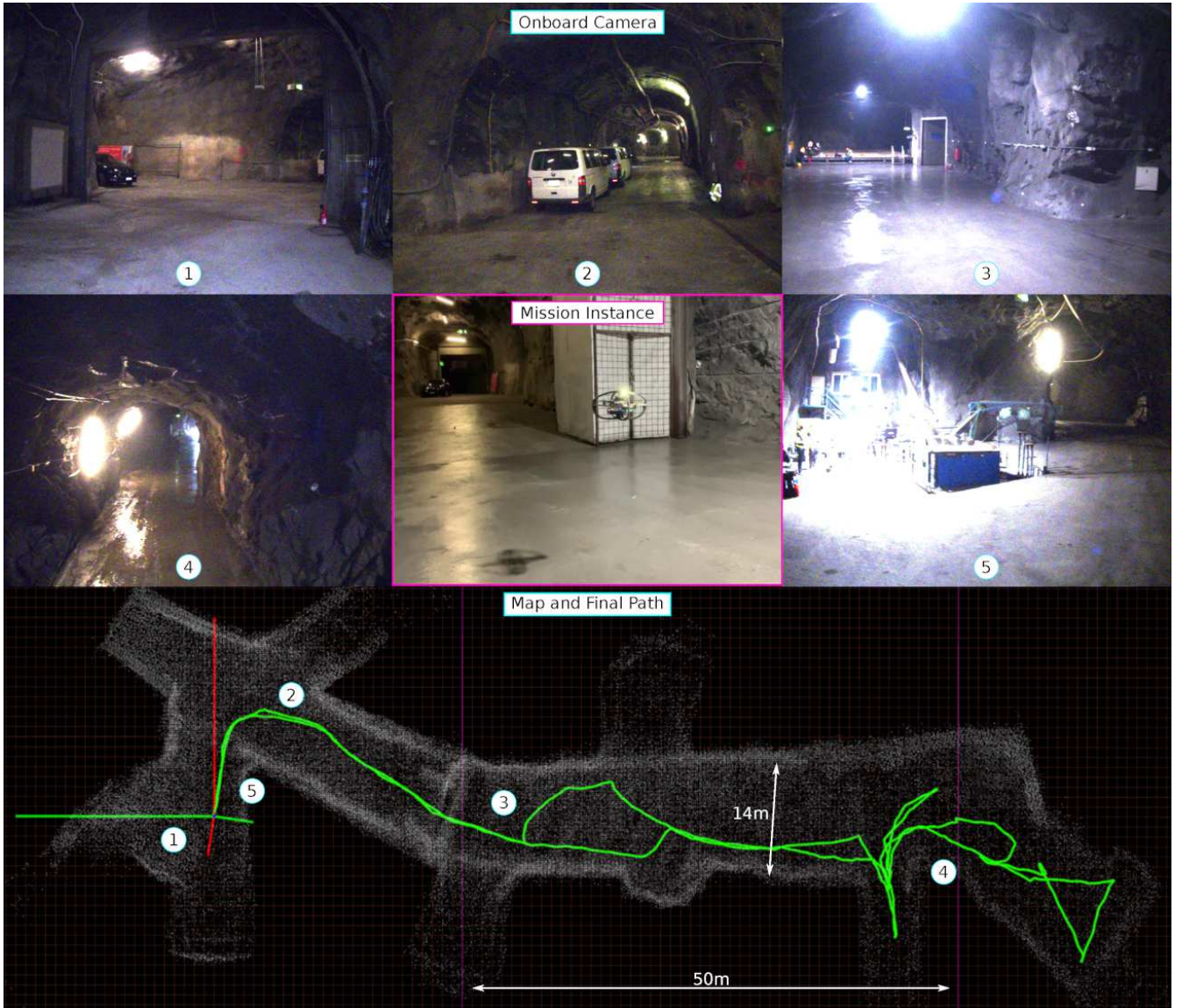


Fig. 9. Instances of an autonomous exploration mission inside the Hagerbach Mine. (1)-(5) instances from the onboard camera. Central image: RMF-Owl in flight. Bottom row: generated map, final executed path and corresponding location of the upper images. Starting from a section of the environment chosen as control station of operation, the robot is able to autonomously takeoff (1), turn right into the largest branch (2), enter the main room (3) and extensively explore a complete dark section of the mine (4) until the homing path is triggered. Finally RMF-Owl safely lands at the takeoff position (5).

to the narrow nature of the environment and to permit better tracking performance, the maximum speed was set to 1.0m/s.

### B. Versuchsstollen Hagerbach

The second field test was conducted in the Versuchsstollen Hagerbach Test Gallery, located in Flums, Switzerland. In contrast with the previous field test environment, in this case the overall dimensions are considerably bigger, especially if compared to the size of RMF-Owl. As can be seen in Figure 9, the biggest section reaches 14m in width. In this second case, the robot also has no prior knowledge of the environment and it is commanded to explore and come back to the takeoff location before the end of its endurance. Starting from the section chosen as main control station, the robot is able to autonomously takeoff (1), turn right into the largest branch (2), enter the main room (3) and extensively explore a complete dark section of the mine

(4) until the homing path is triggered (5). Finally RMF-Owl safely lands at the takeoff position (6). Also in this case the total travelled distance is more than 200m and the total flight time is 8.1min, with maximum speed set to 1.2m/s.

## VI. CONCLUSIONS AND FUTURE WORK

This paper presented the design and overall autonomy functionalities of the RMF-Owl. Tailored to subterranean exploration, the system emphasizes collision-tolerance and resilient autonomy with robust localization and mapping and high-performance exploration path planning in confined, obstacle-filled, and topologically-complex underground environments. Simultaneously, RMF-Owl offers the capacity to search, detect and localize objects of interest which can be proven particularly useful in the framework of search and rescue missions. A set of results from field experiments are presented and allow to demonstrate the capabilities of the system to explore unknown subterranean settings.



## REFERENCES

- [1] T. Dang, M. Tranzatto, S. Khattak, F. Mascarich, K. Alexis, and M. Hutter, "Graph-based subterranean exploration path planning using aerial and legged robots," *Journal of Field Robotics*, vol. 37, no. 8, pp. 1363–1388, 2020.
- [2] C. Papachristos, S. Khattak and K. Alexis, "Autonomous exploration of visually-degraded environments using aerial robots," in *International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2017.
- [3] I. Sa and P. Corke, "Vertical infrastructure inspection using a quadcopter and shared autonomy control," in *Field and service robotics*. Springer, 2014, pp. 219–232.
- [4] A. Bircher, K. Alexis, U. Schwesinger, S. Omari, M. Burri, and R. Siegwart, "An incremental sampling-based approach to inspection planning: the rapidly exploring random tree of trees," *Robotica*, vol. 35, no. 6, pp. 1327–1340, 2017.
- [5] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. L. Grix, F. Ruess, M. Suppa, and D. Burschka, "Toward a fully autonomous uav: Research platform for indoor and outdoor urban search and rescue," *IEEE robotics & automation magazine*, vol. 19, no. 3, pp. 46–56, 2012.
- [6] Defense Advanced Research Projects Agency, "DARPA Subterranean Challenge." [Online]. Available: <https://subtchallenge.com/>
- [7] A. Briod, P. Kornatowski, J.-C. Zufferey, and D. Floreano, "A collision-resilient flying robot," *Journal of Field Robotics*, vol. 31, no. 4, pp. 496–509, 2014.
- [8] C. J. Salan, K. Tadokuma, Y. Okada, Y. Sakai, K. Ohno, and S. Tadokoro, "Development and experimental validation of aerial vehicle with passive rotating shell on each rotor," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2568–2575, 2019.
- [9] S. Atay, M. Bryant, and G. Buckner, "The spherical rolling-flying vehicle: Dynamic modeling and control system design," *Journal of Mechanisms and Robotics*, vol. 13, no. 5, p. 050901, 2021.
- [10] M. Yamada, M. Nakao, Y. Hada, and N. Sawasaki, "Development and field test of novel two-wheeled uav for bridge inspections," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2017, pp. 1014–1021.
- [11] P. D. Petris, H. Nguyen, M. Kulkarni, F. Mascarich, and K. Alexis, "Resilient collision-tolerant navigation in confined environments," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 2286–2292.
- [12] S. Sabet, A.-A. Agha-Mohammadi, A. Tagliabue, D. S. Elliott, and P. E. Nikravesh, "Rollocopter: An energy-aware hybrid aerial-ground mobility for extreme terrains," in *2019 IEEE Aerospace Conference*. IEEE, 2019, pp. 1–8.
- [13] J.-S. Moon, C. Kim, Y. Youm, and J. Bae, "Uni-copter: A portable single-rotor-powered spherical unmanned aerial vehicle (uav) with an easy-to-assemble and flexible structure," *Journal of Mechanical Science and Technology*, vol. 32, no. 5, pp. 2289–2298, 2018.
- [14] Z. Liu and K. Karydis, "Toward impact-resilient quadrotor design, collision characterization and recovery control to sustain flight after collisions," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 183–189.
- [15] S. Mintchev, S. de Rivaz, and D. Floreano, "Insect-inspired mechanical resilience for multicopters," *IEEE Robotics and automation letters*, vol. 2, no. 3, pp. 1248–1255, 2017.
- [16] A. Klapotcz, A. Briod, L. Daler, J.-C. Zufferey, and D. Floreano, "Euler spring collision protection for flying robots," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1886–1892.
- [17] P. Sareh, P. Chermprayong, M. Emmanuelli, H. Nadeem, and M. Kovac, "Rotorigami: A rotary origami protective system for robotic rotorcraft," *Science Robotics*, vol. 3, no. 22, 2018.
- [18] P. M. Kornatowski, S. Mintchev, and D. Floreano, "An origami-inspired cargo drone," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 6855–6862.
- [19] P. Sareh, P. Chermprayong, M. Emmanuelli, H. Nadeem, and M. Kovac, "The spinning cyclic 'miura-oring' for mechanical collision-resilience," *Origami* 7, vol. 3, pp. 981–994, 2018.
- [20] J. Zha, X. Wu, J. Kroeger, N. Perez, and M. W. Mueller, "A collision-resilient aerial vehicle with icosahedron tensegrity structure," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 1407–1412.
- [21] R. de Azambuja, H. Fouad, Y. Bouteiller, C. Sol, and G. Beltrame, "When being soft makes you tough: A collision-resilient quadcopter inspired by arthropods' exoskeletons," *arXiv preprint arXiv:2103.04423*, 2021.
- [22] A. Klapotcz, G. Boutinard-Rouelle, A. Briod, J.-C. Zufferey, and D. Floreano, "An indoor flying platform with collision robustness and self-recovery," in *2010 IEEE International Conference on Robotics and Automation*. IEEE, 2010, pp. 3349–3354.
- [23] C. Vourtsis, W. Stewart, and D. Floreano, "Robotic elytra: Insect-inspired protective wings for resilient and multi-modal drones," *IEEE Robotics and Automation Letters*, 2021.
- [24] H. V. Phan and H. C. Park, "Mechanisms of collision recovery in flying beetles and flapping-wing robots," *Science*, vol. 370, no. 6521, 2020.
- [25] Y. Chen, S. Xu, Z. Ren, and P. Chirarattananon, "Collision resilient insect-scale soft-actuated aerial robots with high agility," *IEEE Transactions on Robotics*, 2021.
- [26] J. Zha and M. W. Mueller, "Exploiting collisions for sampling-based multicopter motion planning," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 7943–7949.
- [27] M. L. Mote, J.-P. Afman, and E. Feron, "A framework for collision-tolerant optimal trajectory planning of autonomous vehicles," *arXiv preprint arXiv:1611.07608*, 2016.
- [28] C. Liu and R. Tron, "Sensing via collisions: a smart cage for quadrotors with applications to self-localization," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021.
- [29] T. Lew, T. Emmei, D. D. Fan, T. Bartlett, A. Santamaria-Navarro, R. Thakker, and A.-a. Agha-mohammadi, "Contact inertial odometry: Collisions are your friends," *arXiv preprint arXiv:1909.00079*, 2019.
- [30] Y. Mulgaonkar, W. Liu, D. Thakur, K. Daniilidis, C. J. Taylor, and V. Kumar, "The tiercel: A novel autonomous micro aerial vehicle that can map the environment by flying into obstacles," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 7448–7454.
- [31] Y. Mulgaonkar, A. Makineni, L. Guerrero-Bonilla, and V. Kumar, "Robust aerial robot swarms without collision avoidance," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 596–603, 2017.
- [32] F. Mascarich, P. De Petris, H. Nguyen, and K. Alexis, "Autonomous distributed 3d radiation field estimation for nuclear environment characterization," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021.
- [33] S. Khattak, H. Nguyen, F. Mascarich, T. Dang, and K. Alexis, "Complementary multi-modal sensor fusion for resilient robot pose estimation in subterranean environments," in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2020, pp. 1024–1029.
- [34] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time," in *Robotics: Science and Systems*, vol. 2, no. 9, 2014.
- [35] S. Lynen, M. Achtelik, S. Weiss, M. Chli, and R. Siegwart, "A robust and modular multi-sensor fusion approach applied to mav navigation," in *Proc. of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS)*, 2013.
- [36] H. Oleynikova, Z. Taylor, M. Fehr, R. Siegwart, and J. Nieto, "Voxblox: Incremental 3d euclidean signed distance fields for on-board mav planning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.
- [37] M. Kulkarni, M. Dharmadhikari, M. Tranzatto, S. Zimmermann, V. Reijgwart, P. De Petris, H. Nguyen, N. Khedekar, C. Papachristos, L. Ott, R. Siegwart, M. Hutter, and K. Alexis, "Autonomous teamed exploration of subterranean environments using legged and aerial robots," *arXiv preprint arXiv:2111.06482*, 2021.
- [38] N. Michael, J. Fink, and V. Kumar, "Cooperative manipulation and transportation with aerial robots," *Autonomous Robots*, vol. 30, 2011.
- [39] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv*, 2018.
- [40] M. Tranzatto, F. Mascarich, L. Bernreiter, C. Godinho, M. Camurri, S. M. K. Khattak, T. Dang, V. Reijgwart, J. Loeje, D. Wisth, S. Zimmermann, H. Nguyen, M. Fehr, L. Solanka, R. Buchanan, M. Bjelonic, N. Khedekar, M. Valceschini, F. Jenelten, M. Dharmadhikari, T. Homberger, P. De Petris, L. Wellhausen, M. Kulkarni, T. Miki, S. Hirsch, M. Montenegro, C. Papachristos, F. Tresoldi, J. Carius, G. Valsecchi, J. Lee, K. Meyer, X. Wu, J. Nieto, A. Smith, M. Hutter, R. Siegwart, M. Mueller, M. Fallon, and K. Alexis, "Cerberus: Autonomous legged and aerial robotic exploration in the tunnel and urban circuits of the darpa subterranean challenge," *Journal of Field Robotics*, 2021.