IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 14, 2020, accepted January 7, 2021, date of publication January 20, 2021, date of current version January 28, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3052967

Autonomous Firefighting Inside Buildings by an Unmanned Aerial Vehicle

VOJTECH SPURNY, VACLAV PRITZL, VIKTOR WALTER, MATEJ PETRLIK, TOMAS BACA, (Graduate Student Member, IEEE), PETR STEPAN, DAVID ZAITLIK, 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)

This work was supported in part by the Czech Science Foundation (GAČR) under Project 20-29531S, in part by the Research Center for Informatics Project under Grant CZ.02.1.01/0.0/0.0/16_019/0000765, in part by CTU under Grant SGS20/174/OHK3/3T/13, in part by the EU H2020 Project AERIAL CORE under Grant 871479, and in part by the Khalifa University via sponsorship in support of 15 selected teams for the MBZIRC 2020 competition.

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 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

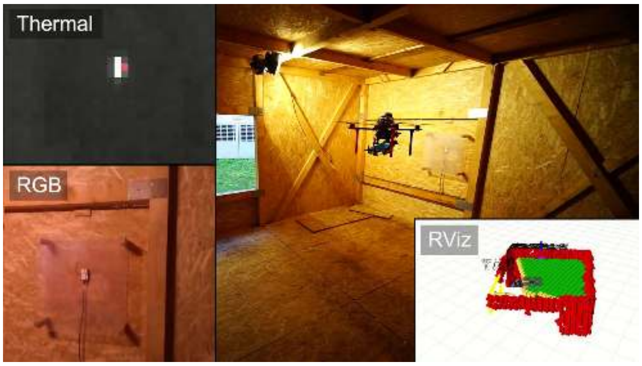


FIGURE 1. The proposed system uses processed data from the onboard sensors of the MAV towards the goal to extinguish fire inside the building.

The associate editor coordinating the review of this manuscript and approving it for publication was Juan Liu.

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 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

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

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 fish-eye 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

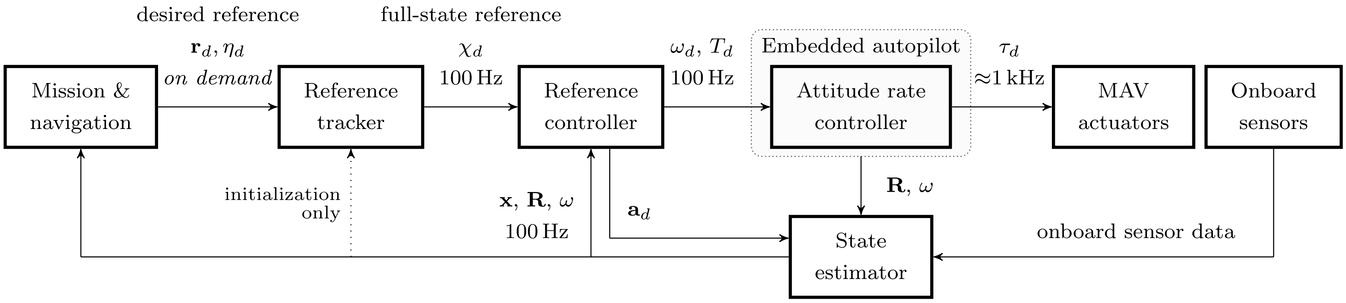


FIGURE 2. **A diagram of the control system architecture. *Mission & navigation* software supplies a position and heading reference ($\mathbf{r}_d$, η_d) to the reference tracker. The *Reference tracker* creates a smooth and feasible reference χ_d for the reference feedback controller. The feedback *Reference controller* produces the desired thrust and angular velocities (T_d, ω_d) for the embedded *Attitude rate controller*. The *State estimator* fuses control input $\mathbf{a}_d$ with data from the onboard sensors to create an estimate of the MAV translation and rotation ($\mathbf{x}$, $\mathbf{R}$, 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

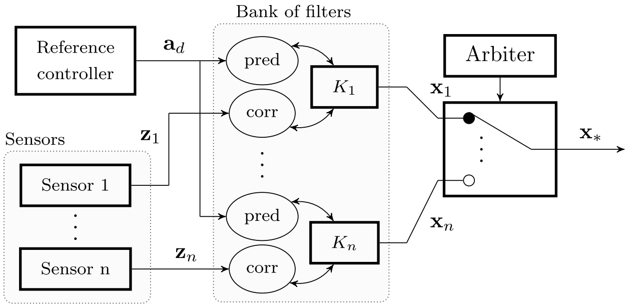


FIGURE 3. **The bank of filters $K = \{K_1,\ldots,K_n\}$. The prediction step is driven by the desired acceleration $\mathbf{a}_d$. The correction step is triggered asynchronously as sensor measurements $\mathbf{z}_i$, where $i = \{1,\ldots,n\}$, arrives. The output hypothesis $\mathbf{x}_*$ is chosen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

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.

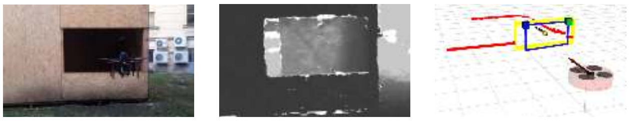

(a) View from an external camera.
(b) Depth image from the onboard camera.
(c) Visualization of the data.

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.**

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
function Detect_windows_depth($\mathcal{I}_{raw}$)
    $\mathcal{I}_{down} \leftarrow$ Downsample($\mathcal{I}_{raw}$)
    $\mathcal{C} \leftarrow$ Find_contours($\mathcal{I}_{down}$)
    for $c \in \mathcal{C}$ do
        $q \leftarrow$ Fit_quadrilateral($c$)
        if $q \neq \varnothing$ and Is_hole($q$) then
            Add $q$ to $\mathcal{D}_{depth}$
    return $\mathcal{D}_{depth}$    ▷ Detections are passed to the LKF
```

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
function Detect_windows($\mathcal{P}$)
    $\mathcal{E}_1, \mathcal{S}_1 \leftarrow$ SEF($\mathcal{P}$)    ▷ Successive Edge Following
    $\mathcal{E}_2, \mathcal{S}_2 \leftarrow$ IEPF($\mathcal{S}_1, \mathcal{P}$)    ▷ Iterative End-Point Fit
    $\mathcal{E} \leftarrow \mathcal{E}_1, \mathcal{E}_2$
    if $mode = depth + lidar$ then
        $\mathcal{D}_{lidar} \leftarrow$ Link_edges_to_estimates($\mathcal{E}, \mathcal{W}$)
    else if $mode = lidar + apriori$ then
        $\mathcal{D}_{lidar} \leftarrow$ Standalone_detection($\mathcal{E}, \mathcal{A}$)
    return $\mathcal{D}_{lidar}$    ▷ 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 $\mathbf{x}$ of the Kalman filter describing a single window is defined as

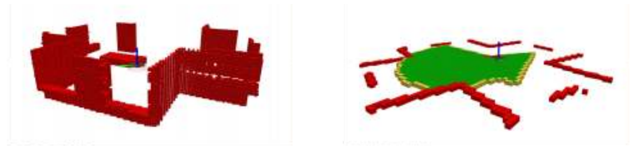
$$\mathbf{x} = \left[c_x, c_y, c_z, \phi, w, h\right]^T , \tag{1}$$

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 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 × 128 × 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



(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.

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 $0.3\,\mathrm{m\,s^{-1}}$ and a heading rate of $0.3\,\mathrm{rad\,s^{-1}}$. This is within the capabilities of the MAV platform that had flying dynamic constraints set to a speed of $2\,\mathrm{m\,s^{-1}}$ and a acceleration of $10\,\mathrm{m\,s^{-2}}$. 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).

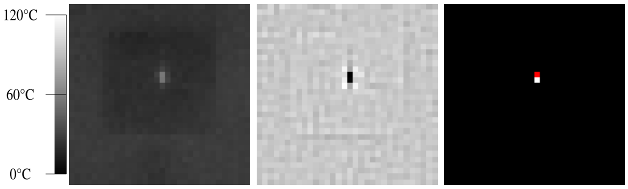


FIGURE 6. Thermal vision outputs. From left to right: Thermal camera view scaled from 0 to 120 °C, differential image produced with the Laplacian operator (scaled between the two extreme values of the image), and thresholded detection with the rounded centroid in red.

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 $\epsilon$
function Explore_interior($\epsilon$)
    $\mathcal{B}_{plan} \leftarrow [\,]$                                   ▷ initialize 3D grid for planning
    $\mathcal{B}_{expl} \leftarrow [\,]$                                   ▷ initialize 2D grid for exploration
    while not Fire_detected() do
        $\mathcal{R} \leftarrow$ Get_current_state_of_MAV()                ▷ $\mathcal{R} = (\mathbf{x}, \mathbf{R})$
        if Received_new_data_from_sensors() then                 ▷ From the stereo camera or 2D LIDAR
            $d_{new} \leftarrow$ Get_new_data()
            $\mathcal{B}_{plan} \leftarrow$ Update_planning_buffer($\mathcal{B}_{plan}, d_{new}, \mathcal{R}$)
            $\mathcal{B}_{expl} \leftarrow$ Update_exploration_buffer($\mathcal{B}_{expl}, d_{new}, \mathcal{R}, \epsilon$)
        if Is_time_to_replan() then
            $f \leftarrow$ Get_nearest_frontier($\mathcal{B}_{expl}, \mathcal{R}$)
            if Is_empty($f$) then
                return False                                     ▷ Space has been explored without detection
            $Plan \leftarrow$ Plan_trajectory($\mathcal{B}_{plan}, \mathcal{R}, f, \epsilon$)   ▷ $Plan = [\mathbf{r}_1, \eta_1, \ldots, \mathbf{r}_n, \eta_n]$
            Fly_trajectory($Plan$)
    Hover()                                                      ▷ Stop following the previous trajectory
    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:

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

where

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

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 $\hat{\mathbf{v}}_\mathbf{f}$.

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

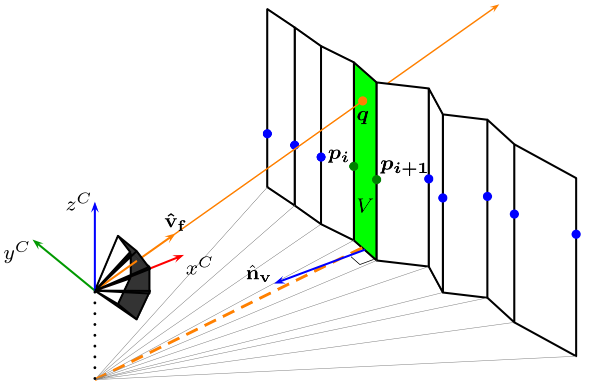


FIGURE 7. Illustration of how the fire is localized.

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 $\boldsymbol{p_i}$ and $\boldsymbol{p_{i+1}}$ from $\mathcal{P}_{lidar}$, s.t.

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

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

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 $\boldsymbol{p_i}$ 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 $\boldsymbol{q}$ is then calculated as:

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

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

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

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

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

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

$$\mathbf{x}_i = \left[c_{i_x}, c_{i_y}, c_{i_z}, \psi_i\right]^T, \tag{10}$$

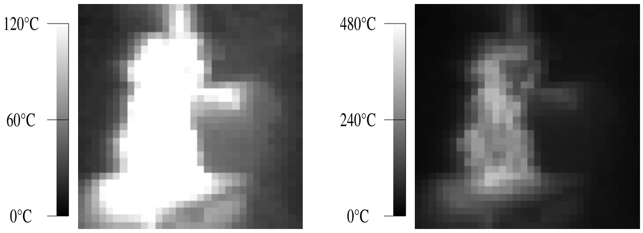


FIGURE 8. An example of the view of a real fire with the thermal camera used in the competition at two different thermal ranges. Note the high contrast of the fire compared to the background, in addition to the large size of the fires in the image. In the range of temperatures used in the competition, 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.

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 $\boldsymbol{q}$ and normal $\mathbf{n_v}$, if $\boldsymbol{q}$ is closer in the world frame than 1 m to $\left[c_{i_x}, c_{i_y}, c_{i_z}\right]^T$ 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 $\boldsymbol{s}$ 1.5 m in front of the given target $\boldsymbol{q}$ 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)).

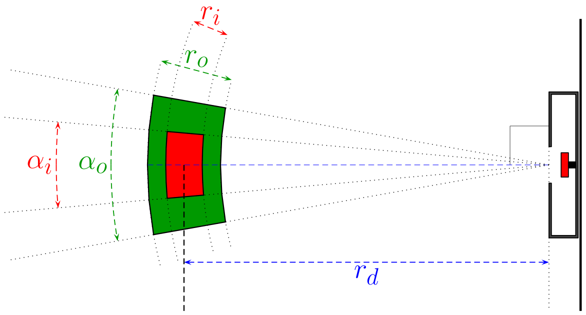


FIGURE 9. Steering hysteresis for fire extinguishing, viewed from above.

In this state, the MAV is steered towards position $\boldsymbol{s}$. 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

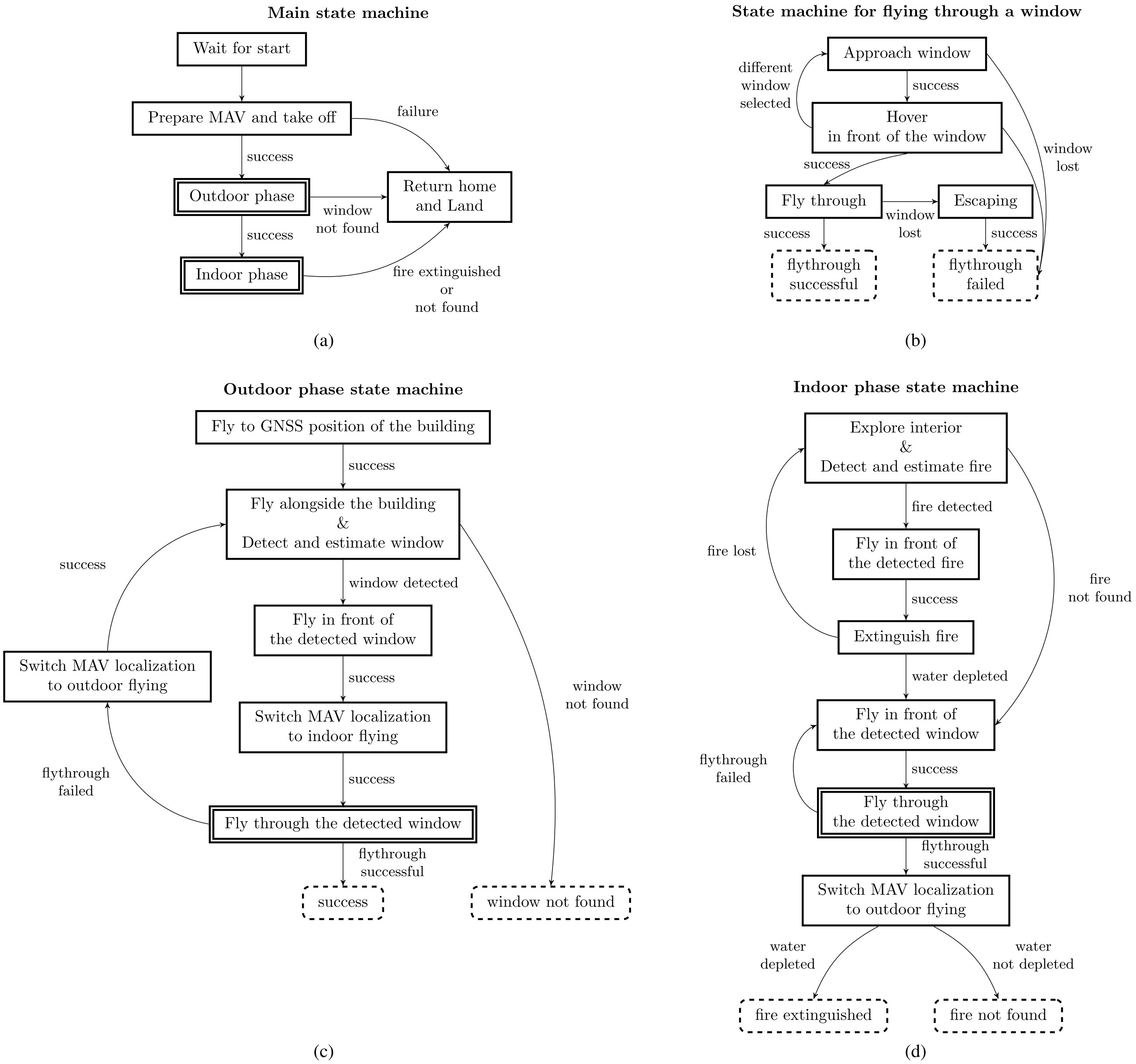

Main state machine
Wait for start
Prepare MAV and take off
failure
success
Outdoor phase
Return home and Land
window not found
success
Indoor phase
fire extinguished or not found
(a)
State machine for flying through a window
Approach window
different window selected
success
Hover in front of the window
window lost
success
Fly through
Escaping
window lost
success
success
flythrough successful
flythrough failed
(b)
Outdoor phase state machine
Fly to GNSS position of the building
success
Fly alongside the building & Detect and estimate window
success
window detected
Fly in front of the detected window
Switch MAV localization to outdoor flying
success
window not found
Switch MAV localization to indoor flying
flythrough failed
success
Fly through the detected window
flythrough successful
success
window not found
(c)
Indoor phase state machine
Explore interior & Detect and estimate fire
fire detected
fire lost
Fly in front of the detected fire
fire not found
success
Extinguish fire
water depleted
Fly in front of the detected window
flythrough failed
success
Fly through the detected window
flythrough successful
Switch MAV localization to outdoor flying
water depleted
water not depleted
fire extinguished
fire not found
(d)

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

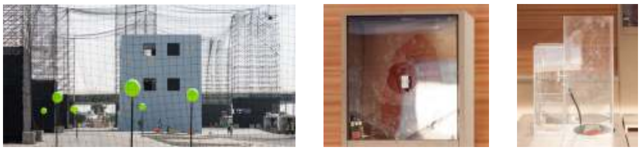


FIGURE 11. **Photos of the MBZIRC 2020 Fire Challenge area. The photo on the left shows the tall structure simulating a building, the photo in the middle shows front view of the fire analogues (this unit was turned on) and the photo on the right shows side view of the fire analogues (this unit was turned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

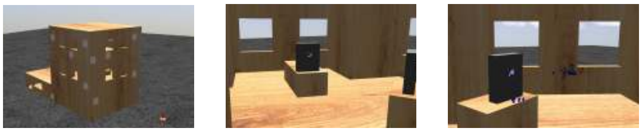


FIGURE 12. **Snapshots from the simulation developed for the Fire Challenge of the MBZIRC 2020 competition.**

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 m × 60 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 m × 2 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

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?	✓	×	×	✓	✓	✓	✓	✓	✓	✓

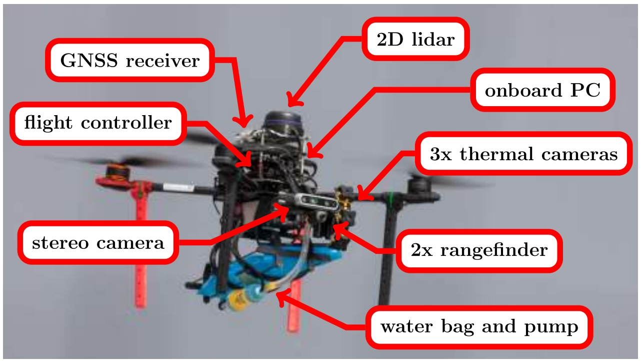


FIGURE 13. Description of the components of the deployed MAV platform for indoor fire extinguishing.

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) $87^\circ \pm 3^\circ \times 58^\circ \pm 1^\circ \times 95^\circ \pm 3^\circ$ 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 resolution 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

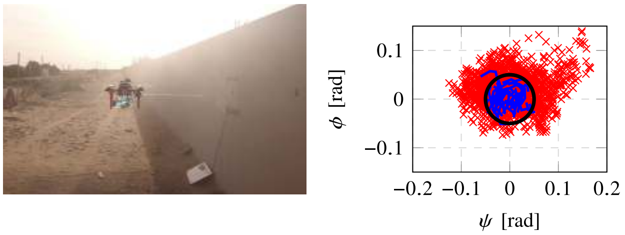


FIGURE 14. Experimental verification of fire extinguishing in the desert near Abu Dhabi, United Arab Emirates. The image on the left shows a photo from this experiment. The image on the right displays the plot of the changing offsets of the nozzle direction from the estimated fire analogue direction 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].

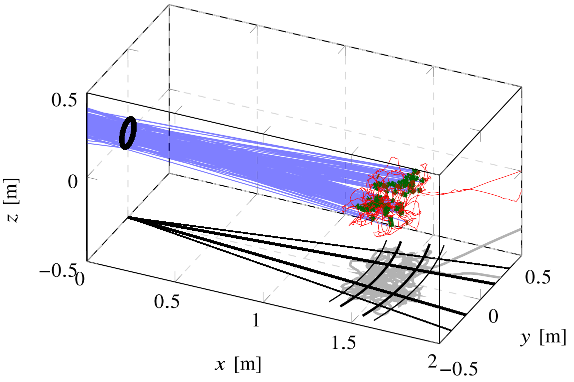


FIGURE 15. Plot of the MAV trajectory during the extinguishing experiment displayed in Figure 14. The red line denotes the trajectory itself relative to the fire analogue. The green points are specific positions from which the MAV activated its water pump. Spraying lines parallel to the direction of 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.

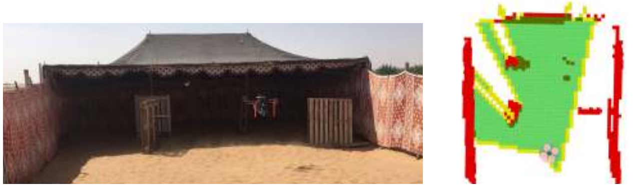
FIRE DETECTION AND EXTINGUISHING

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

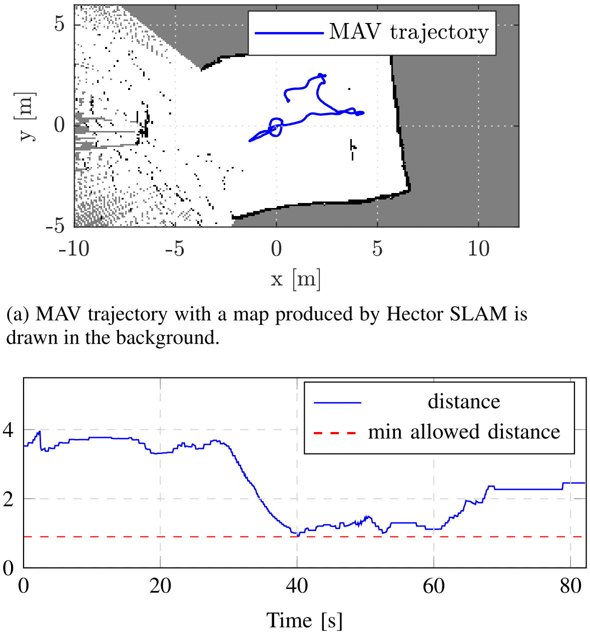
Another experimentally verified subsystem was indoor motion planning and exploration (discussed in section II-C). The goal of the experiment was to completely explore the space inside a room with obstacles consisting of poles holding the structure and wooden artificial obstacles. An example layout of 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



(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.

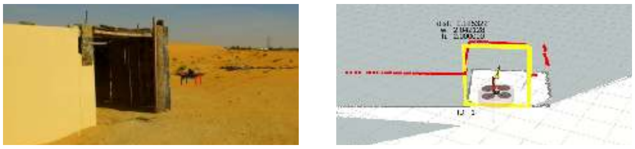
(b) Distance of the MAV from obstacles during the flight.

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 system. 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.



(a) MAV in front of the mock-up building. (b) Visualization of sensor data.

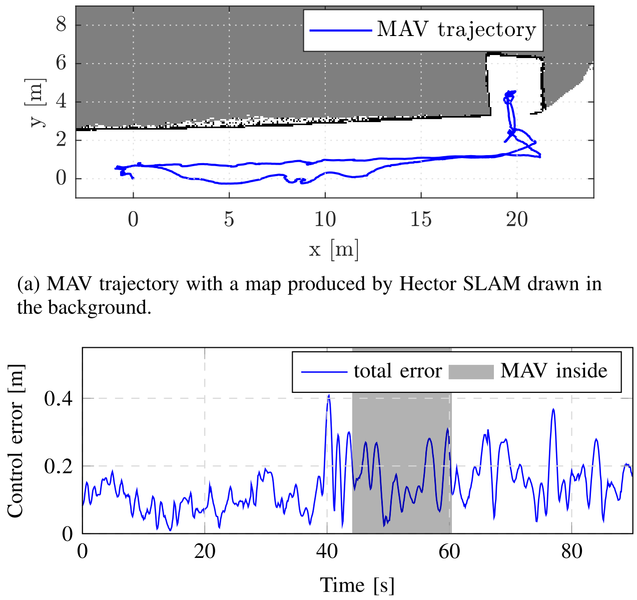
FIGURE 18. Images from experimental verification of the correct performance of outdoor wall-following, window detection, flight through the detected window, and switching between indoor and outdoor modes of localization. 18(a) shows the MAV before entering a mock-up building in the desert and 18(b) displays 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].

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

[12] https://youtu.be/9LTf6PG4ijc

[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.

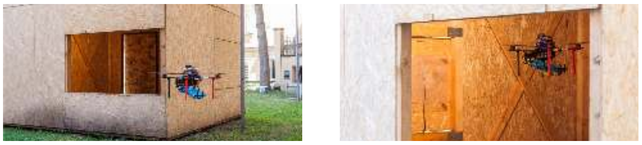
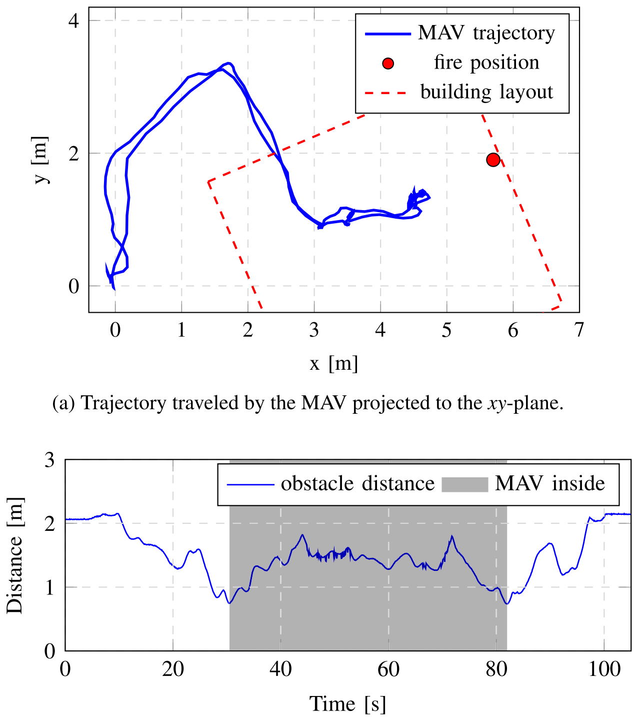


FIGURE 20. Experimental verification of the proposed system on a mock-up of the building set up in the Czech Republic. The image on the left shows the MAV outside the mock-up during its search for an open window. The image on the right displays the MAV while the water is being sprayed on the fire analogue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



(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.

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 firefighting 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

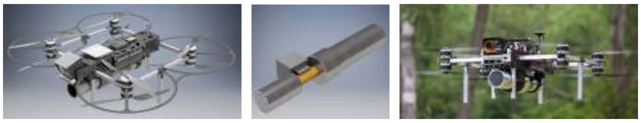


FIGURE 22. A new platform under development that uses fire-extinguishing capsules as projectiles. The first two photos on the right show 3D render of the platform and launcher design, photo on the left shows constructed platform.

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 recurrent neural networks,'' in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2018, pp. 1788–1791.

[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. Petrlik, 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.

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.

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.

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.

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.

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.

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.